AD-A041 383 B-K DYNAMICS INC SAN DIEGO CALIF F/G 5/9
DOCUMENTATION ON THE FAST MODEL UTILITIES.(U)
APR 77 E H ROWLEY N00014-76-C-0726
UNCLASSIFIED BKD-TR-3-231 NL

| OF |
AD
A041383

END
DATE
FILMED
7 - 77

12

12

TR-3-231

DOCUMENTATION ON THE FAST MODEL UTILITIES

1 April 1977

B-K DYNAMICS, INC.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 14 BKD TR-3-231	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) 6 DOCUMENTATION ON THE FAST MODEL UTILITIES,		5. TYPE OF REPORT & PERIOD COVERED Technical Report 7-1-76 to 3-31-77
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) 10 Edward H. Rowley		8. CONTRACT OR GRANT NUMBER(s) 15 N00014-76-C-0726
9. PERFORMING ORGANIZATION NAME AND ADDRESS B-K Dynamics, Inc. 2727 Camino Del Rio South, Suite 219 San Diego, CA 92108		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS ONR (Code 434)		12. REPORT DATE 11 1 April 1977
		13. NUMBER OF PAGES 12 65p.
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) 9 Technical rept. 1 Jul 76 - 31 Mar 77,		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release;

Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The FAST Utilities are a family of processes which constitute a key element of the Force Analysis and Simulation Technique Model (FAST). The forecasted gains and losses to inventory within the Navy enlisted personnel force structure are derived within the FAST Utilities. These forecasted gains and losses are the product of independently derived prediction coefficients and corresponding base inventories. Within the FAST Utilities, the forecasts are also subject to user options, management directed alternatives, integerization, internal normalization, and inner-consistency constraints.

410267

FOREWARD

The FAST Utilities are a family of processes which constitute a key element of the Force Analysis and Simulation Technique Model (FAST). The forecasted gains and losses to inventory within the Navy enlisted personnel force structure are derived within the FAST Utilities. These forecasted gains and losses are the product of independently derived prediction coefficients and corresponding base inventories. The forecast derivation process is further refined by provisions for model user options and management directed alternatives within the FAST Utilities. Finally, the forecasts are subject to integerization, internal normalization, and inner-consistency constraints.

TABLE OF CONTENTS

1.0 Purpose and Scope

The Analytic Treatise of the FAST Utilities Model is intended for use by the Personnel Managers in the Bureau of Naval Personnel, by the FAST technical analysts, and by students of force structure simulation techniques. It delineates the algorithms implemented within the model and provides the means for determining the reaction of the model to a given input condition or set of conditions. The treatise does not attempt to recount the motivating rationale behind the model, nor does it attempt to provide an exacting description of the techniques necessary to the computerized implementation of the model. It does provide a detailed account of the mathematical and logical processes within the model. The treatise assumes that the reader is generally familiar with:

(a) the framework of the Navy enlisted personnel system and force structure,

(b) the basic design and use of the Navy Force Analysis and Simulation Technique Model (FAST).

2.0 Introduction

2.1 General

The FAST Utilities may be generally thought of as a family of modules within the FAST model. The FAST Utilities provide essential services to the operation of the FAST model. These services include:

(1) the derivation of the forecast force sturcture from the base force structure and the appropriate prediction coefficient,

(2) the facilitation of management manipulation of the force structure by means of user directed alternatives,

(3) the imposition of rational constraints on the model and on the management directed alternatives, and

(4) the integerization of the force structure and resolution of inconsistencies within the force structure.

2.2 Enlisted Personnel Structure Characterization

The FAST Utilities perform operations on matrices which represent flows within the enlisted personnel system. This system may be characterized as groups of three-dimensional arrays which classify all personnel or personnel-related parameters by paygrade, length of service, and rating. Each group represents data pertaining to a given forecast type, where the forecast type relates the data elements associated with a given type of gain or loss to inventory. Rating is the accepted nomenclature for the nearly one-

hundred occupational specialties within the enlisted force. The rating plane of an array is the basic processing unit in the FAST Utilities and categorizes the rating by thirty-one (31) length of service (LOS) and seven (7) paygrade categories. The seven paygrade categories correspond to one aggregate row containing the sum of the three enlisted apprenticeship paygrades and six additional rows correspond to the six descrete petty officer paygrades. An eighth row is added to this dimension of the array to contain the paygrade vector total over the LOS columns. The thirty-one (31) LOS columns correspond to the thirty-one categories of active service in one year intervals. A thirty-second column is added to provide the LOS vector total over each paygrade row. This enlisted personnel structure is illustrated in Figure 1.

2.3 Subroutines

The FAST Utilities are, in fact, subroutines within the FAST software system and may be referred to by their subroutine names as follows:

(1) MAST
(2) FLPFLP
(3) APPFOR
(4) ROUND
(5) ARAB
(6) ARNG
(7) NORMAL

3.0 Process Flow

The subroutines MAST, APPFOR, ROUND, ARAB, and ARNG are invoked by FAST in sequential order, as illustrated in the flow diagram in Figure 2. Also illustrated is the condition in which MAST invokes FLPFLP when warranted. NORMAL is invoked by ARNG with each pass through the Utilities.

The process from MAST through ARAB performs operations upon a single rating plane with a given forecast type, as diagrammed in Figure 2. This process is repeated for each rating in the forecast type before the system flow continues to ARNG. The entire three dimensional data array for a given forecast type becomes a consideration in ARNG. ARNG invokes NORMAL to process each rating vector of the array until the entire array has been processed. FAST Utilities conclude processing of the array for a given forecast type upon completion of ARNG. The FAST system uses the utilities to process each forecast type to complete the forecast of the enlisted personnel structure.

4.0 Subroutine Summaries

4.1 MAST and FLPFLP Summary

The individual subroutines of the FAST Utilities each fulfill unique objectives. MAST derives the forecast array for a given rating and forecast type by applying the prediction coefficient array for that rating and forecast type to the appropriate base inventory array. MAST allows that the current values within this initial forecast array may not conform to the defined relationships between interior elements and vector total elements. MAST imposes conformance to these relationships while attempting to preserve the form

or contour of the initial forecast array. MAST provides four user directed alternative methods to impose conformance. The most straight-forward is acceptance of the current interior cells of the array and direct replacement of the current paygrade totals in row 8 and the LOS totals in column 32 by the summations of the current interior paygrade and LOS vectors. The three remaining alternatives involve user selection of either the current paygrade totals, the current LOS totals, or the current grand total as the desired component. MAST invokes FLPFLP to project the selected component inward to establish the defined relationships throughout the array, while holding the selected component constant.

4.2 APPFOR Summary

The APPFOR subroutine accomodates management directed changes to the force structure by providing a technique whereby exogenous values may be imposed upon any cell or combination of cells within the forecast array. APPFOR maintains the defined relationships between components of the array by making appropriate adjustments to accomodate the imposed value.

4.3 ROUND Summary

The ROUND subroutine acknowledges the need to provide enlisted force data in terms of 'whole men' and integerizes the cells of the forecast array which were, thus far, mixed numbers. ROUND maintains the defined relationships between components of the array by making appropriated adjustments as the rounding-off of the various components of the array occur.

4.4 ARAB Summary

The ARAB subroutine fulfills the system constraint provides that no cell of the rounded forecast array may be greater than the corresponding cell of the base inventory array from which it was derived. ARAB continues to maintain integer values and the defined relationship between components of the array.

4.5 ARNG and NORMAL Summary

The ARNG subroutine is a monitor routine which reconciles inconsistancies between the summations of cells through rating vectors in the three-dimensional enlisted personnel structure and corresponding cells in the all-navy prediction array. See Figure 1 for an illustration of a depthwise rating vector. ARNG invokes NORMAL to reconcile each rating vector, until all rating vectors in the structure have been processed. The total vectors are then reestablished by direct summation of the appropriate interior vectors.

The detailed descriptions of the algorithm implemented by each FAST Utility subroutine follows.

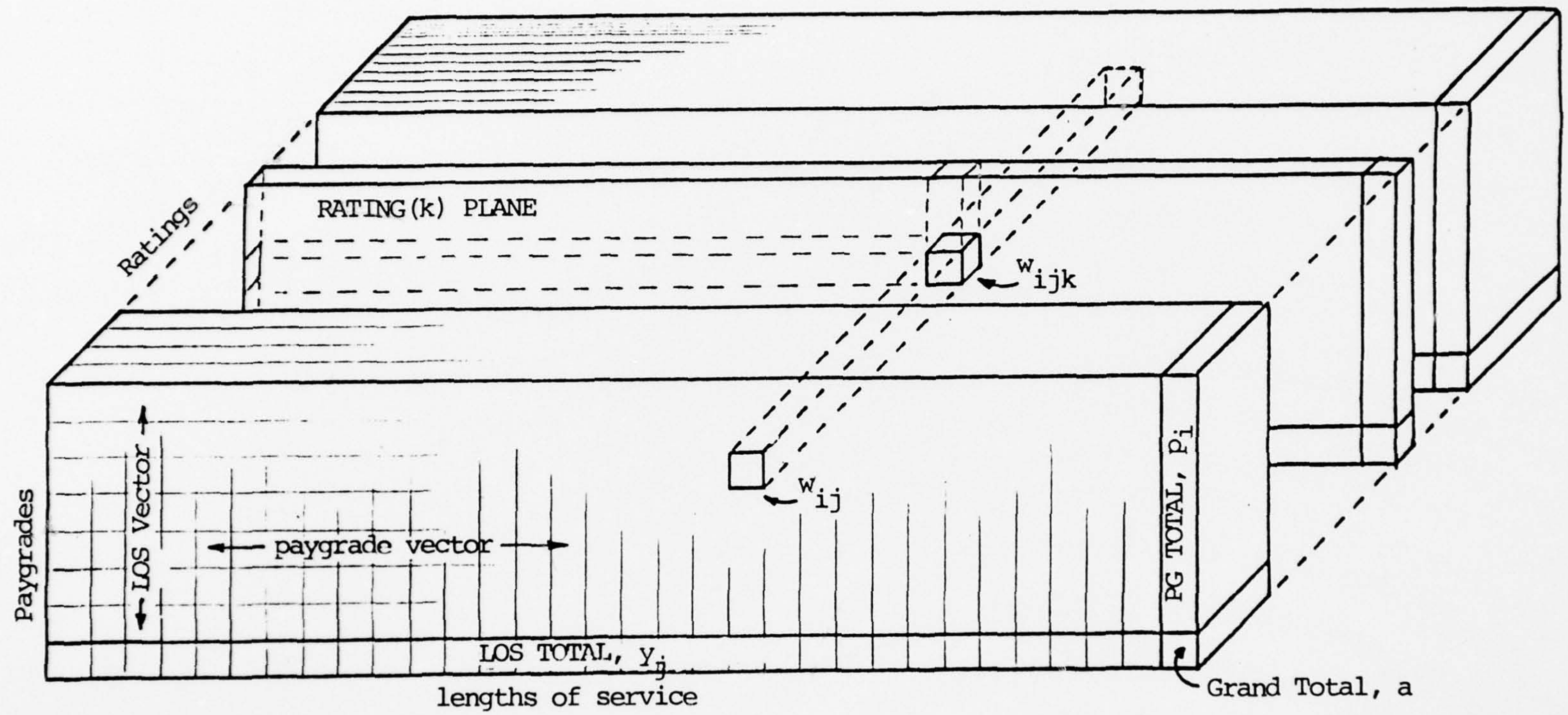


Figure 1

Enlisted Personnel Structure Characterization

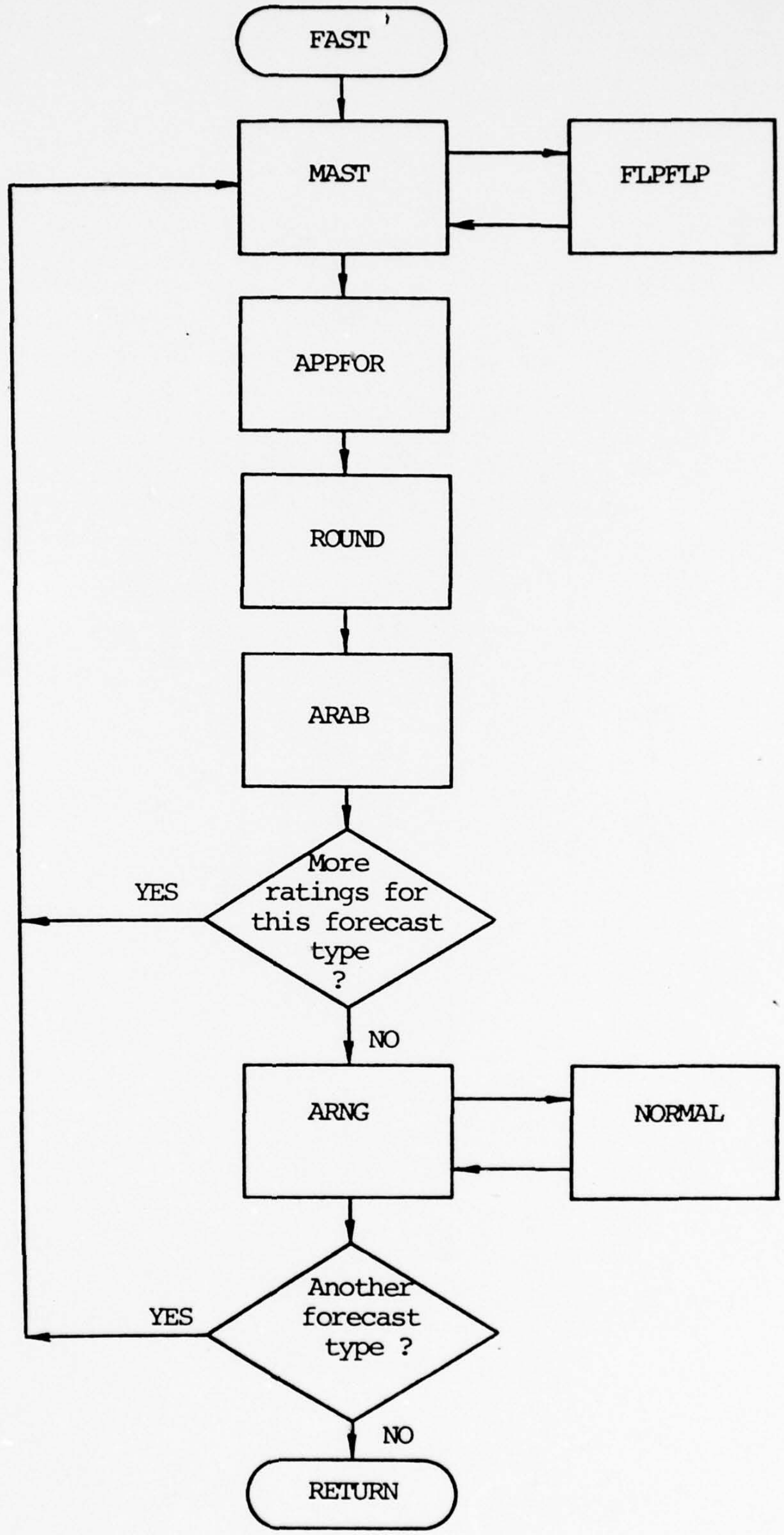


Figure 2. FAST Utilities System Flow

APPENDIX A

MAST

B&K
DYNAMICS, INC.
15825 SHADY GROVE ROAD
ROCKVILLE, MARYLAND 20850

MAST

The initial computation of the MAST subroutine applies an independently derived prediction coefficient array to the appropriate base array. This derives the forecast array containing predicted losses (or gains) to the base. The base array to be operated upon is inherent to the program. The computation is performed cell-by-cell to arrays classified by paygrade and length of service for a given loss (or gain) type within a given rating. This may be expressed as:

$$w_{ij}^{(t)} = \alpha_{ij} \cdot s_{ij}^{(t-1)} \qquad \{1\}$$

where, $w_{ij}^{(t)}$ = the forecast array containing predicted number of losses (or gains) from the base array, with paygrade i, length of service j, during period t.

α_{ij} = the independently derived, historically smoothed, prediction coefficient for paygrade i, and length of service j.

$s_{ij}^{(t-1)}$ = the base array with paygrade i, length of service j, at the end of period t-1, (thus interpreted as beginning of period t).

and i = paygrade level ($i = 1,2,\ldots,7$) and $i=8$ representing the LOS total vector.

j = length of service ($j = 1,2,\ldots,31$) and $j=32$ representing the PG total vector

All further computation within the FAST Utilities represent occurrances within the time period t, therefore, the notation of t will be dropped and considered as implied.

The distinction between interior cells of the forecast array and those cells representing the LOS total vector, paygrade total vector, and grand total becomes important in further processes. This distinction will be emphasized by distinguishing mnemonics, as follows:

Paygrade Total Vector Cells, become p_i, where:

$$p_i = \sum_{j=1}^{31} w_{ij} \qquad \text{and } i = 1,2,\ldots,7$$

LOS Total Vector Cells, become y_j, where:

$$y_j = \sum_{i=1}^{7} w_{ij} \qquad \text{and } j = 1,2,\ldots,31$$

Grand Total Cell, becomes a, where:

$$a = \sum_{i=1}^{7} \sum_{j=1}^{31} w_{ij}$$

Interior Cells, remain as w_{ij}, and $i = 1,2,\ldots,7$, $j = 1,2,\ldots,31$

While tracking the progress of a cell or cells through a subroutine, the content of that cell is frequently redefined by the process. Superscript notation will be used to denote the successive redefinitions of a parameter in a subroutine. The initial content as it is provided to the subroutine will not be superscripted. Thus, w_{ij}, $w_{ij}^{(1)}$, $w_{ij}^{(2)}$, and $w_{ij}^{(3)}$ represent the value of an interior cell as it enters the subroutine and progresses through three successive redefinitions. An alphabetic superscript is used occasionally to represent temporary computational values of array components which do not redefine the component.

An array is said to be in "accord" when the following conditions exist simultaneously:

$$w_{i,32} = p_i, \qquad \text{for } i = 1,2,\ldots,7,$$

$$w_{8,j} = y_j, \qquad \text{for } j = 1,2,\ldots,31,$$

and $w_{8,32} = a.$

Once MAST has derived its initial forecast, it allows that the components of the resulting array may not be in accord. MAST is, thus, further tasked with inducing accord between these components. MAST recognizes four conceivable alternatives to induce accord. Each alternative acknowledges the validity of a different component of the initial array and proceeds to induce changes in the remaining components of the array to establish accord. The alternative components are:

(1) Interior Cells, w_{ij}

(2) LOS Total Vector, y_j

(3) PG Total Vector, p_i

(4) Grand Total, a.

The selection of alternative is an empirically established parameter to the program and causes a branch to the appropriate procedure. Upon completion of any one of the alternative procedures, MAST executes a return to the calling program with the forecasted array in accord. The alternative procedures are as follows:

(1) Interior Cells, w_{ij}

The selection of interior cells, w_{ij}, as the basis from which to develop accord within the array, results in a straightforward summation of the interior vectors to compute replacements for the existing PG total vector cells, LOS total vector cells, and grand total cell. This is accomplished by a call to the SUMMAT subroutine.

The SUMMAT subroutine first clears the content of all components of the array, other than the interior, by setting p_i, a, and y_j, equal to 0.0. The routine then computes the sum of each PG interior vector, the sum of each LOS interior vector, and the aggregate of all interior cells. These sums may be expressed as:

$$p_i^{(1)} = \sum_{j=1}^{31} w_{ij} \quad , \quad i = 1,2,\ldots,7 \tag{2}$$

$$y_j^{(1)} = \sum_{i=1}^{7} w_{ij} \quad , \quad j = 1,2,\ldots,31 \tag{3}$$

$$a^{(1)} = \sum_{j=1}^{31} \sum_{i=1}^{7} w_{ij} \tag{4}$$

SUMMAT then returns to MAST, and MAST returns to the calling program. The output forecast array of MAST, when summing from interior cells, is represented by w_{ij}, $p_i^{(1)}$, $y_j^{(1)}$, and $a^{(1)}$.

(2) LOS Total Vector, y_j

The selection of the LOS total vector cells, y_j, as the basis from which to develop accord within the array, proceeds by first redefining the grand total cell as the sum of the current LOS total vector, i.e.:

$$a^{(1)} = \sum_{j=1}^{31} y_j \tag{5}$$

This desired grand total is then apportioned to the PG total vector. This is accomplished by first computing the current sum of the existing PG total vector, $a^{(tp)}$, which represents a temporary computational value of the grand total. $a^{(tp)}$ does not redefine $a^{(1)}$.

$$a^{(tp)} = \sum_{i=1}^{7} p_i \tag{6}$$

The apportionment of the desired grand total to the PG total vector is then completed by applying the ratio of the desired to existing grand total to the existing PG total cells, therefore, redefining the cells as:

$$p_i^{(1)} = p_i \cdot (a^{(1)}/a^{(tp)}) \,, \qquad \text{for } i = 1,2,\ldots,7 \qquad \{7\}$$

The MAST subroutine has, thus, established accord among the values of y_j, $a^{(1)}$, and $p_i^{(1)}$, which are next passed to the FLPFLP subroutine along with the original interior cells, w_{ij}. The FLPFLP subroutine attempts to apportion these totals among the interior cells by means of a converging iteration. The objective of FLPFLP is to accomplish total accord among the components of the array. FLPFLP is described as an independent FAST Utility.

(3) PG Total Vector, p_i

The selection of the PG total vector cells, p_i, as the basis from which to develop accord within the array proceeds by first redefining the grand total as the sum of the PG total vector, or

$$a^{(1)} = \sum_{i=1}^{7} p_i \qquad \{8\}$$

This desired grand total is then apportioned to the LOS total vector. This is accomplished by first computing the current sum of the existing LOS total vector, $a^{(ty)}$, which represents a temporary computational value of the grand total. $a^{(ty)}$ does not redefine $a^{(1)}$.

$$a^{(ty)} = \sum_{j=1}^{31} y_j \qquad \{9\}$$

The apportionment of the desired grand total to the LOS total vector is then completed by applying the ratio of the desired to existing grand total to the existing LOS total cells, therefore redefining the cells as:

$$y_j^{(1)} = y_j \cdot (a^{(1)}/a^{(ty)}), \quad \text{for } j = 1,2,\ldots,31 \qquad \{10\}$$

The MAST subroutine has, thus, established accord among the values of p_i, $a^{(1)}$, and $y_j^{(1)}$, which are next passed to the FLPFLP subroutine along with the original interior cells, w_{ij}. As with the LOS total vector procedure, the FLPFLP subroutine attempts to apportion these totals among the interior cells by means of a converging iteration. The objective of FLPFLP is to accomplish total accord among the components of the array. FLPFLP is described as an independent FAST Utility.

(4) Grand Total, a

The selection of the grand total cell, a, as the basis from which to develop accord within the array proceeds by first computing the current sum of the existing PG total vector. This temporary computational value of the grand total, $a^{(tp)}$, does not redefine a, but is a value from which an apportionment ratio may be computed.

$$a^{(tp)} = \sum_{i=1}^{7} p_i \qquad \{11\}$$

The apportionment of the desired grand total, a, to the PG total vector is then complete by applying the ratio of the desired to current grand total to the current PG total cells, therefore redefining the p_i as:

$$p_i^{(1)} = p_i \cdot (a/a^{(tp)}), \quad \text{for } i = 1,2,\ldots,7 \qquad \{12\}$$

Another temporary computational value of the grand total, $a^{(ty)}$, is computed from the current LOS total vector. This value, $a^{(ty)}$, does not redefine a.

$$a^{(ty)} = \sum_{j=1}^{31} y_j \qquad \{13\}$$

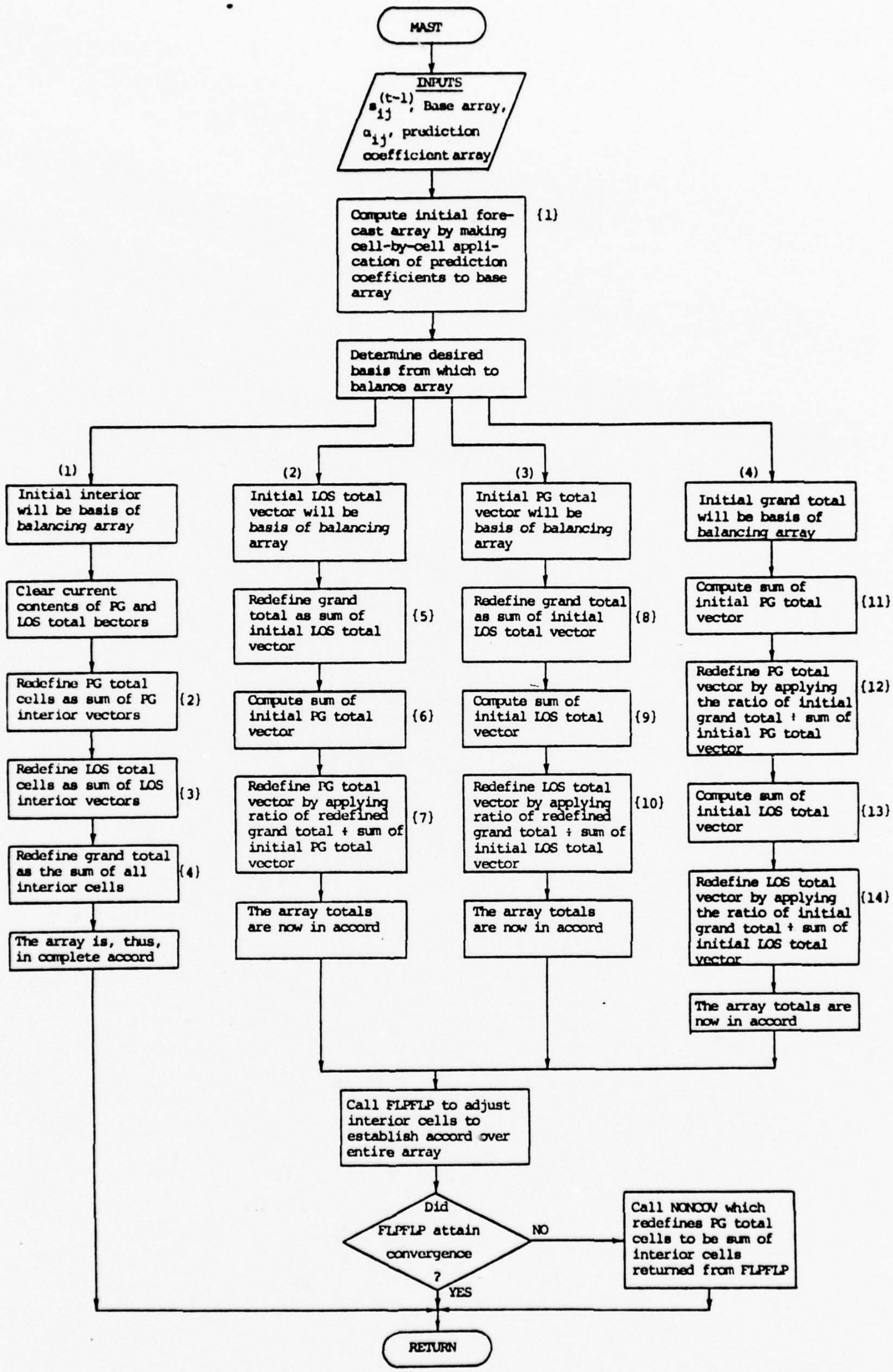
And the desired grand total, a, is apportioned to the LOS total vector by:

$$y_j^{(1)} = y_j \cdot (a/a^{(ty)}), \qquad \text{for } j = 1,2,\ldots,31 \qquad \{14\}$$

The MAST subroutine has, thus, established accord among the values of a, $p_i^{(1)}$, and $y_j^{(1)}$. MAST now calls FLPFLP to attempt to apportion these totals among the interior and achieve total accord by means of a converging iteration. The array defined by a, $p_i^{(1)}$, $y_j^{(1)}$, and w_{ij} is passed on to FLPFLP by MAST.

The subroutine, FLPFLP, is described as an independent FAST Utility.

The return from FLPFLP to MAST is followed by a test to verify convergence by the iterations of FLPFLP. If convergence is confirmed, processing continues by a return to the calling program. If convergence was not attained by FLPFLP, MAST calls the subroutine NONCOV which repairs the nonconverging array by redefining the PG total cells to be the sum of the appropriate interior PG vector cells derived in the final iteration by FLPFLP. NONCOV recognizes that the last unsuccessful iteration in FLPFLP left an LOS total vector which was in accord with the interior. MAST then returns to the calling program.


MAST
INPUTS
$s_{ij}^{(t-1)}$, Base array,
α_{ij}, prediction coefficient array
Compute initial forecast array by making cell-by-cell application of prediction coefficients to base array
(1)
Determine desired basis from which to balance array
(1)
Initial interior will be basis of balancing array
Clear current contents of PG and LOS total bectors
Redefine PG total cells as sum of PG interior vectors
(2)
Redefine LOS total cells as sum of LOS interior vectors
(3)
Redefine grand total as the sum of all interior cells
(4)
The array is, thus, in complete accord
(2)
Initial LOS total vector will be basis of balancing array
Redefine grand total as sum of initial LOS total vector
(5)
Compute sum of initial PG total vector
(6)
Redefine PG total vector by applying ratio of redefined grand total ÷ sum of initial PG total vector
(7)
The array totals are now in accord
(3)
Initial PG total vector will be basis of balancing array
Redefine grand total as sum of initial LOS total vector
(8)
Compute sum of initial LOS total vector
(9)
Redefine LOS total vector by applying ratio of redefined grand total ÷ sum of initial LOS total vector
(10)
The array totals are now in accord
(4)
Initial grand total will be basis of balancing array
Compute sum of initial PG total vector
(11)
Redefine PG total vector by applying the ratio of initial grand total ÷ sum of initial PG total vector
(12)
Compute sum of initial LOS total vector
(13)
Redefine LOS total vector by applying the ratio of initial grand total ÷ sum of initial LOS total vector
(14)
The array totals are now in accord
Call FLPFLP to adjust interior cells to establish accord over entire array
Did FLPFLP attain convergence ?
NO
YES
Call NONCOV which redefines PG total cells to be sum of interior cells returned from FLPFLP
RETURN

APPENDIX B

FLPFLP

B&K
DYNAMICS, INC.
15825 SHADY GROVE ROAD
ROCKVILLE, MARYLAND 20850

FLPFLP

The FLPFLP subroutine of FAST is called by MAST when it desires to project balanced total vectors inward to establish accord among the interior cells of a prediction array. The desired total cells, a, p_i, and y_j, as they are received from MAST, are in accord with each other. The interior cells received from MAST, w_{ij}, are used by FLPFLP as weighting factors for distributing the total cells to the interior.

The subroutine first forces the interior cells to sum in accord with the paygrade total vector cells. A temporary computational sum, $p_i^{(tw)}$, of the current value of each paygrade interior vector is made. $p_i^{(tw)}$ does not redefine p_i.

$$p_i^{(tw)} = \sum_{j=1}^{31} w_{ij} \qquad \text{for } i = 1,2,\ldots,7 \tag{1}$$

The interior cells are then redefined by a proportional distribution of the desired sum, p_i, to the interior.

$$w_{ij}^{(1)} = w_{ij} \cdot (p_i/p_i^{(tw)}) \tag{2}$$

$$\text{for } i = 1,2,\ldots,7 \quad \text{and} \quad j = 1,2,\ldots,31$$

Subroutine FLPFLP now enters a series of up to 200 iterations during which it attempts to converge upon a condition of total accord among the components of the array. The process performs alternating passes over the LOS vectors and the PG vectors, computing the difference between the desired sum of each vector and the current sum. The difference, if greater than 1.0, is distributed over the vector proportionally to its current content. The iteration terminates and convergance is declared when the process makes a pass (LOS vectors or PG vectors) in which no difference greater than 1.0 is found in any vector during the pass. Non-convergence within 200 iterations results

in a return to MAST, which, consequently, calls the NONCOV subroutine to redefine the PG Total Vector.

The iteration begins by making a pass, in turn, through each of thirty-one LOS vectors. Readjustment along the LOS vector begins by determining the current, temporary, LOS total from the current interior cells.

$$y_j^{(tw)} = \sum_{i=1}^{7} w_{ij}^{(1)} \tag{3}$$

The absolute value of the difference between the desired and the current LOS total cells is computed.

$$d = |y_j - y_j^{(tw)}| \tag{4}$$

If d is less than 1.0, the program proceeds to the next LOS vector. IF d is 1.0 or greater, then an adjustment is made to the interior cells of that vector.

$$w_{ij}^{(2)} = w_{ij}^{(1)} \cdot (y_j/y_j^{(tw)}) \tag{5}$$

$$i = 1,2,\ldots,7$$

If no adjustment was required during the processing of all the LOS vectors, j = 1,2,...31, because no difference of 1.0 or greater was found, the program recognizes this condition as acceptable convergence and FLPFLP returns to MAST. If convergence has not been attained, FLPFLP continues the iteration by making a pass through each of the seven PG vectors.

The readjustment along the PG vector is accomplished in the same manner as that for the LOS vector. The current,

temporary, PG total from the current interior cells is computed.

$$p_i^{(tw)} = \sum_{j=1}^{31} w_{ij}^{(2)} \qquad \{6\}$$

The absolute value of the difference between the desired and the current PG total cells is computed

$$d = |p_i - p_i^{(tw)}| \qquad \{7\}$$

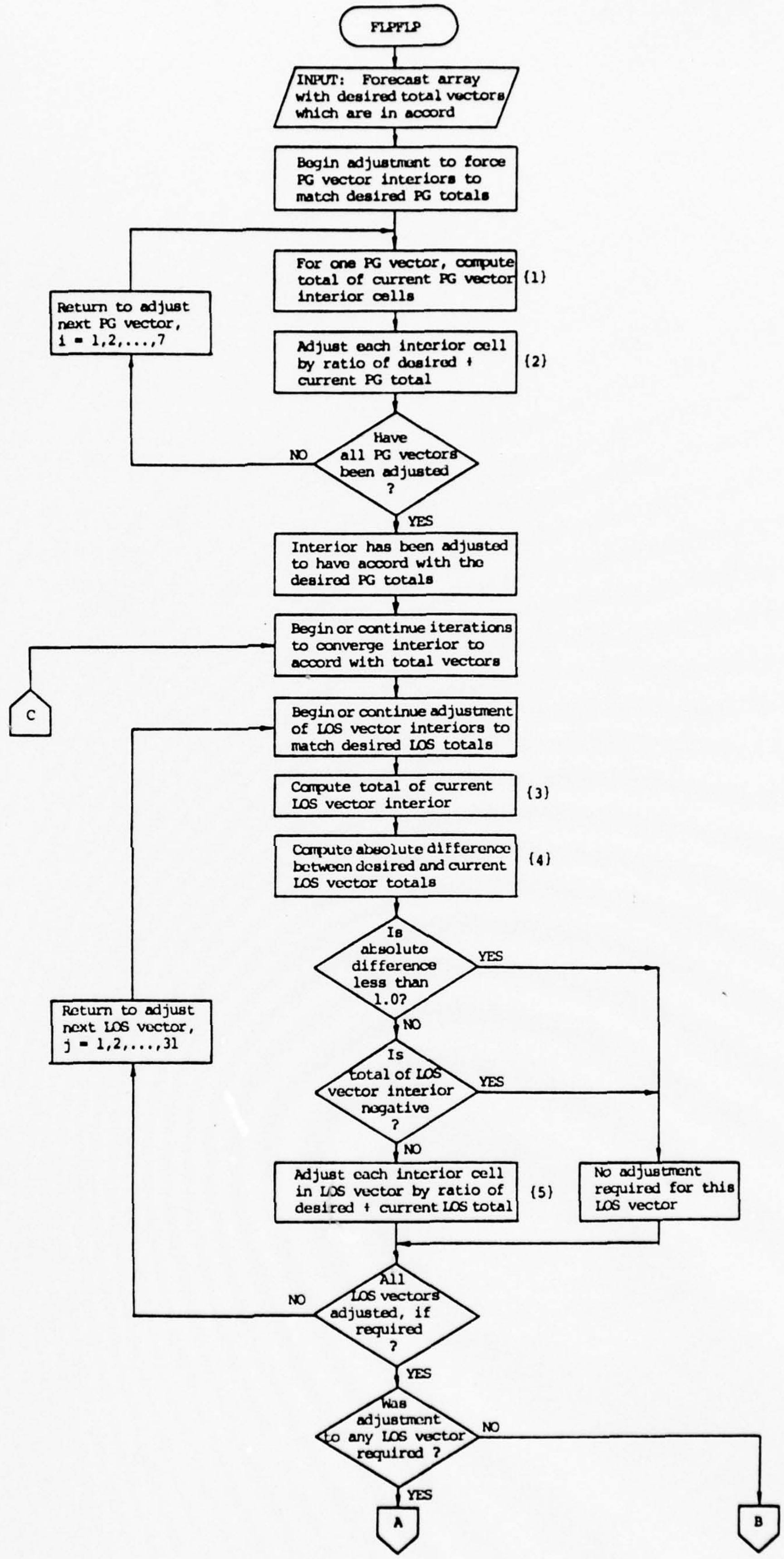
If d is less than 1.0, the program proceeds to the next PG vector. If d is 1.0 or greater, then an adjustment is made to the interior cells of that vector.

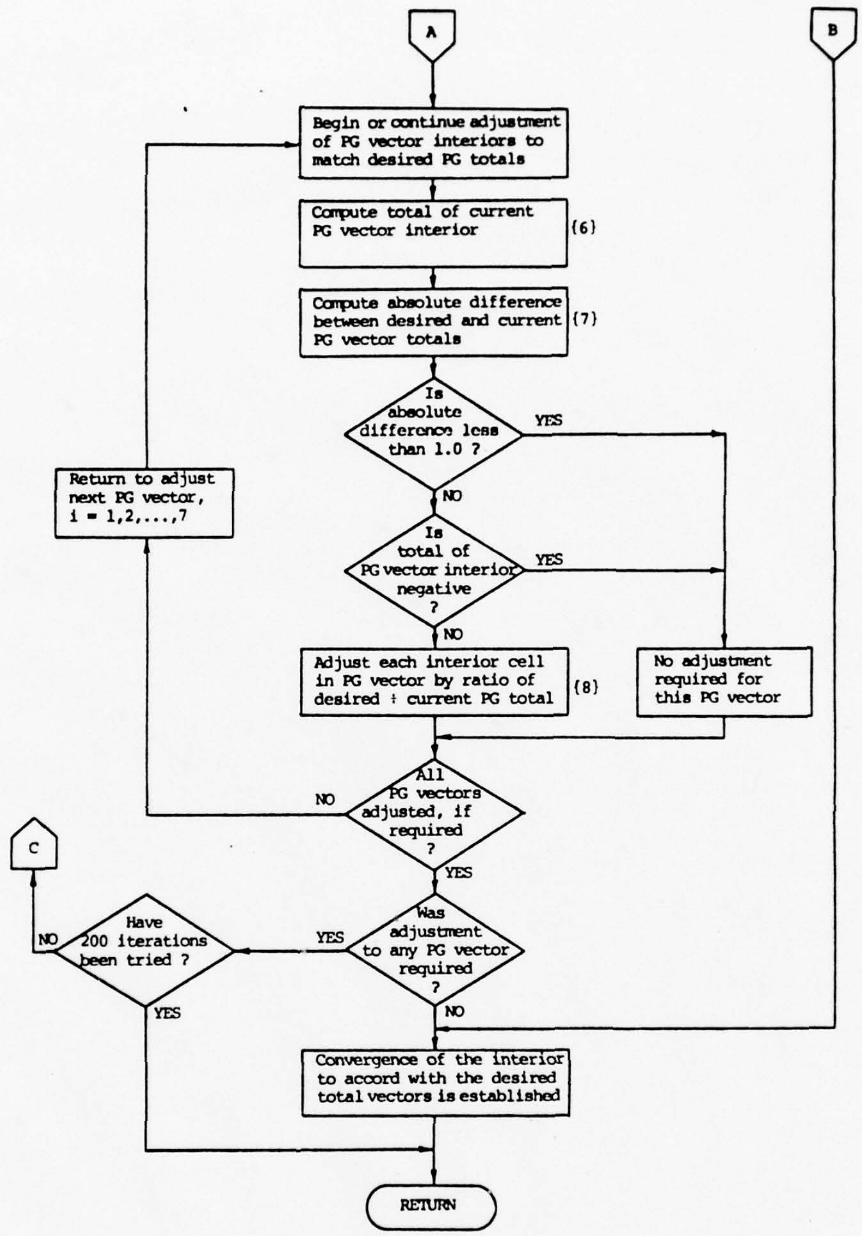
$$w_{ij}^{(3)} = w_{ij}^{(2)} \cdot (p_i/p_i^{(tw)}) \qquad \{8\}$$

$$j = 1,2,\ldots,31$$

If no adjustment was required during the processing of all the PG vectors, i = 1,2,...,7, because no difference of 1.0 or greater was found, the program recognizes this as acceptable convergence and FLPFLP returns to MAST. If convergence has not been attained, FLPFLP continues to the next iteration by returning to the above described pass through the LOS vectors.

As stated earlier, the maximum number of iterations through the LOS and PG vector adjustments is two hundred.


FLPFLP
INPUT: Forecast array with desired total vectors which are in accord
Begin adjustment to force PG vector interiors to match desired PG totals
For one PG vector, compute total of current PG vector interior cells
(1)
Return to adjust next PG vector, i = 1,2,...,7
Adjust each interior cell by ratio of desired ÷ current PG total
(2)
Have all PG vectors been adjusted ?
NO
YES
Interior has been adjusted to have accord with the desired PG totals
Begin or continue iterations to converge interior to accord with total vectors
C
Begin or continue adjustment of LOS vector interiors to match desired LOS totals
Compute total of current LOS vector interior
(3)
Compute absolute difference between desired and current LOS vector totals
(4)
Is absolute difference less than 1.0?
YES
NO
Return to adjust next LOS vector, j = 1,2,...,31
Is total of LOS vector interior negative ?
YES
NO
Adjust each interior cell in LOS vector by ratio of desired ÷ current LOS total
(5)
No adjustment required for this LOS vector
All LOS vectors adjusted, if required ?
NO
YES
Was adjustment to any LOS vector required ?
NO
YES
A
B


A
B
Begin or continue adjustment of PG vector interiors to match desired PG totals
Compute total of current PG vector interior
(6)
Compute absolute difference between desired and current PG vector totals
(7)
Is absolute difference less than 1.0 ?
YES
NO
Is total of PG vector interior negative ?
YES
NO
Adjust each interior cell in PG vector by ratio of desired ÷ current PG total
(8)
No adjustment required for this PG vector
Return to adjust next PG vector, i = 1,2,...,7
All PG vectors adjusted, if required ?
NO
YES
Was adjustment to any PG vector required ?
YES
NO
Have 200 iterations been tried ?
NO
YES
C
Convergence of the interior to accord with the desired total vectors is established
RETURN

APPENDIX C

APPFOR

B&K
DYNAMICS, INC.
15825 SHADY GROVE ROAD
ROCKVILLE, MARYLAND 20850

APPFOR

The APPFOR subroutine accomodates management directed changes to the force structure by providing a technique whereby exogenous values may be imposed upon any cell or combination of cells within the force structure. Such values may reflect existing or proposed policies, or they may represent actual changes occurring to the force structure.

The APPFOR subroutine first processes exogenous inputs to be forced to the interior cells of the forecast array, w_{ij}. The correlative value(s) to be forced into each cell are represented, mnemonically, as f^w_{ij}.

The process of redefining interior cells by force may be expressed as:

$$w^{(1)}_{ij} = \begin{cases} w_{ij} & , \quad \text{if } f^w_{ij} = -1.0 \quad \{1\} \\ w_{ij} \cdot (1 + f^w_{ij}), & \quad \text{if } 0 < |f^w_{ij}| < 1.0 \quad \{2\} \\ f^w_{ij} & , \quad \text{if } f^w_{ij} \geq 1.0 \text{ or } f^w_{ij} = 0 \quad \{3\} \end{cases}$$

Expression {1} represents a condition set to distinguish between zero and blank in the computer and results in no redefinition. Expression {2} recognizes a percentage or proportionate change to the interior cell. The maximum proportionate change is just less than 100%. Expression {3} represents the basic process where a direct substitution of the desired value is made to redefine the interior cell.

Following each redefinition of an individual cell of the forecast array, the corresponding LOS total, y_j, the corresponding PG total, p_i, and, consequently the array grand total must be adjusted appropriately. The adjustment to each is the difference between the original value of the cell and the redefined value, $(w^{(1)}_{ij} - w_{ij})$.

These adjustments, in turn, follow the form:

$$y^{(1)}_j = y_j + (w^{(1)}_{ij} - w_{ij}) \qquad \{4\}$$

$$p^{(1)}_i = p_i + (w^{(1)}_{ij} - w_{ij}) \qquad \{5\}$$

$$a^{(1)} = a + (w^{(1)}_{ij} - w_{ij}) \qquad \{6\}$$

The interior cell forcing process is performed iteratively for i = 1,2,...,7 and, within each value of i, iteratively for j = 1,2,...,31 until all interior cells of the force array have been tested and applied to the cells of the forecast array.

Next in the heirarchy of processing within APPFOR is the iterative test and application of inputs, f^p_i, to be forced to the paygrade total vector, $p^{(1)}_i$, with i = 1,2,...,7.

The process of redefining the PG total cells by force may be expressed as:

$$p^{(2)}_i = \begin{cases} p_i^{(1)} & , \text{ if } f_i^p = -1.0 \qquad \{7\} \\ p_i^{(1)} \cdot (1 + f_i^p) & , \text{ if } 0 < |f_i^p| < 1.0 \qquad \{8\} \\ f_i^p & , \text{ if } f_i^p \geq 1.0 \text{ or } f_i^p = 0 \qquad \{9\} \end{cases}$$

Expression {7} results in no redefinition. Expression {8} recognizes a percentage or proportionate change to the PG total cell, which cannot reach or exceed 100%. Expression {9} represents a direct substitution of the desired value to redefine the current PG total cell.

Following each redefinition of each paygrade total cell, the array grand total is appropriately adjusted by:

$$a^{(2)} = a^{(1)} + (p_i^{(2)} - p_i^{(1)}) \qquad \{10\}$$

The change in the paygrade total is then apportioned to the interior cells of that paygrade vector by:

$$w_{ij}^{(2)} = w_{ij}^{(1)} \cdot (p_i^{(2)} / p_i^{(1)}) \quad \text{for } j = 1,2,\ldots,31 \qquad \{11\}$$

Following each interior cell change, the amount of the change is added to the appropriate LOS total cell by:

$$y_j^{(2)} = y_j^{(1)} + (w_{ij}^{(2)} - w_{ij}^{(1)}) \qquad \{12\}$$

The FORCE subroutine then processes, iteratively, the test and application of inputs, f_j^Y, to be forced to the LOS total vector, $y_j^{(2)}$, with $j = 1,2,\ldots,31$.

The process of redefining the LOS total cells by force may be expressed as:

$$y_j^{(3)} = \begin{cases} y_j^{(2)} & , \text{ if } f_j^Y = -1.0 \qquad \{13\} \\ y_j^{(2)} \cdot (1 + f_j^Y) & , \text{ if } 0 < |f_j^Y| < 1.0 \qquad \{14\} \\ f_j^Y & , \text{ if } f_j^Y \geq 1.0 \text{ or } f_j^Y = 0 \qquad \{15\} \end{cases}$$

Expression {13} results in no redefinition. Expression {14} recognizes a percentage or proportionate change to the LOS total cell, which cannot reach or exceed 100%. Expression {15} represents a direct substitution of the desired value to redefine the current LOS total cell.

The readjustment process parallels the paygrade total force process, using the formulas:

$$a^{(3)} = a^{(2)} + (y_j^{(3)} - y_j^{(2)}) \tag{16}$$

$$w_{ij}^{(3)} = w_{ij}^{(2)} \cdot (y_j^{(3)} / y_j^{(2)}) \tag{17}$$

and $$p_i^{(3)} = p_i^{(2)} + (w_{ij}^{(3)} - w_{ij}^{(2)}) \tag{18}$$

The final process in the APPFOR subroutine processes the highest order of exogenous input, f^a, to be forced to the array grand total, $a^{(3)}$.

This process may be expressed as:

$$a^{(4)} = \begin{cases} a^{(3)} & , \text{ if } f^a = -1.0 & \{19\} \\ a^{(3)} \cdot (1 + f^a), & \text{if } 0 < |f^a| < 1.0 & \{20\} \\ f^a & , \text{ if } f^a \geq 1.0 \text{ or } f^a = 0 & \{21\} \end{cases}$$

Since the grand total has been forced to change, the paygrade total vector, the LOS total vector, and the forecast array interior cells no longer correlate. The cell values are, therefore, reapportioned by application of the ratio of the redefined grand total to the previous grand total, i.e.: $a^{(4)}/a^{(3)}$.

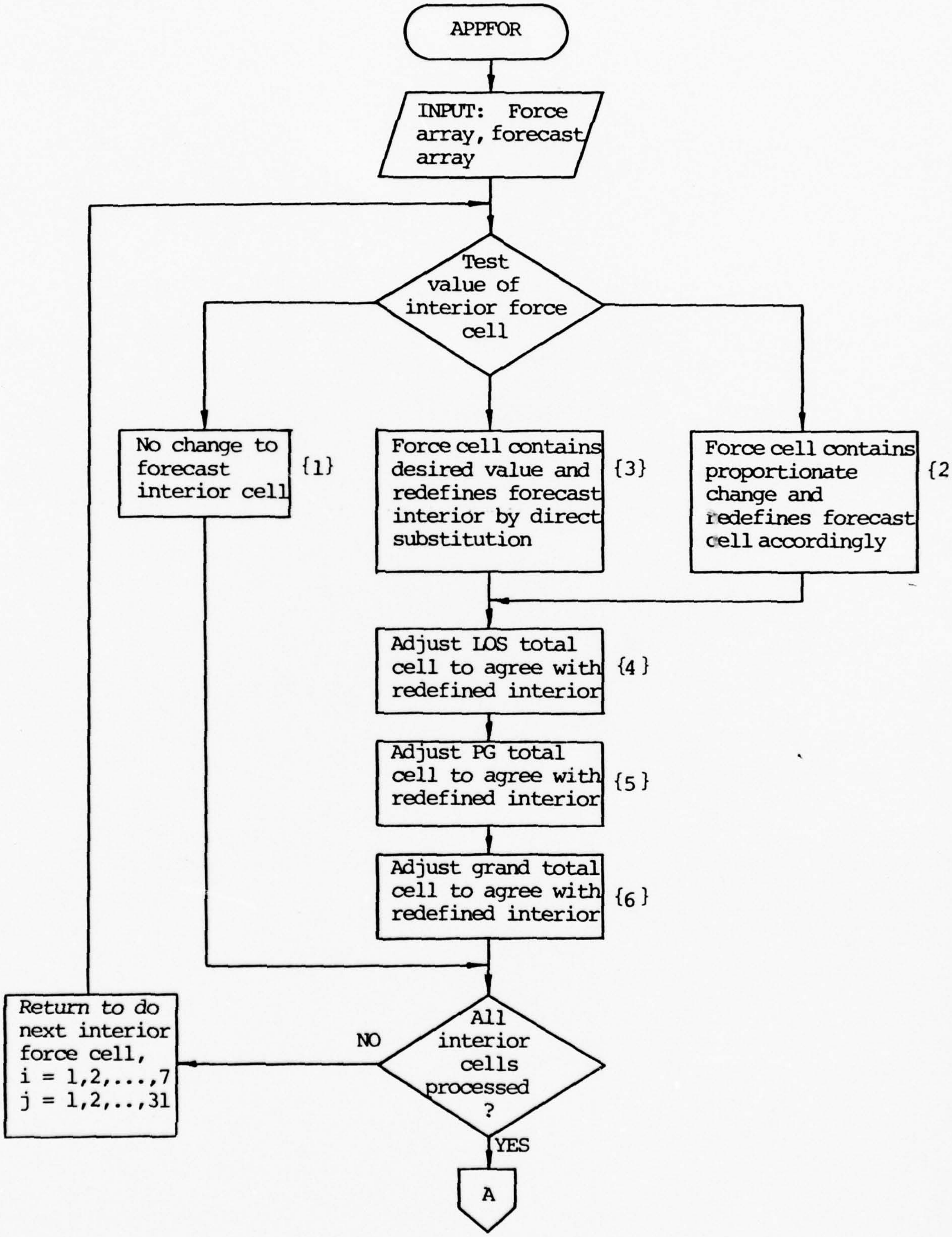
The following formulas illustrate the method of correlation:

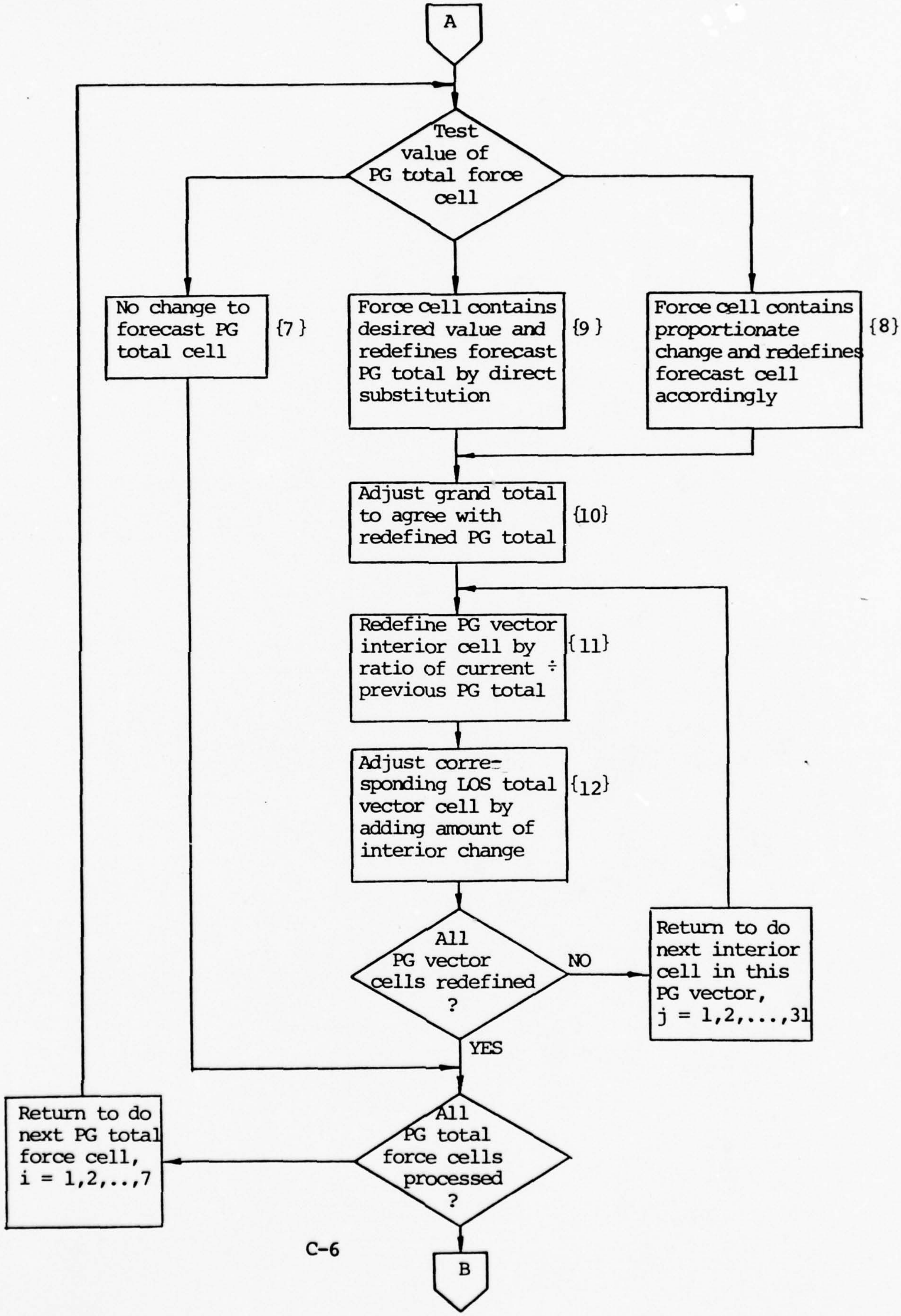
$$w_{ij}^{(4)} = r \cdot w_{ij}^{(3)} \tag{22}$$

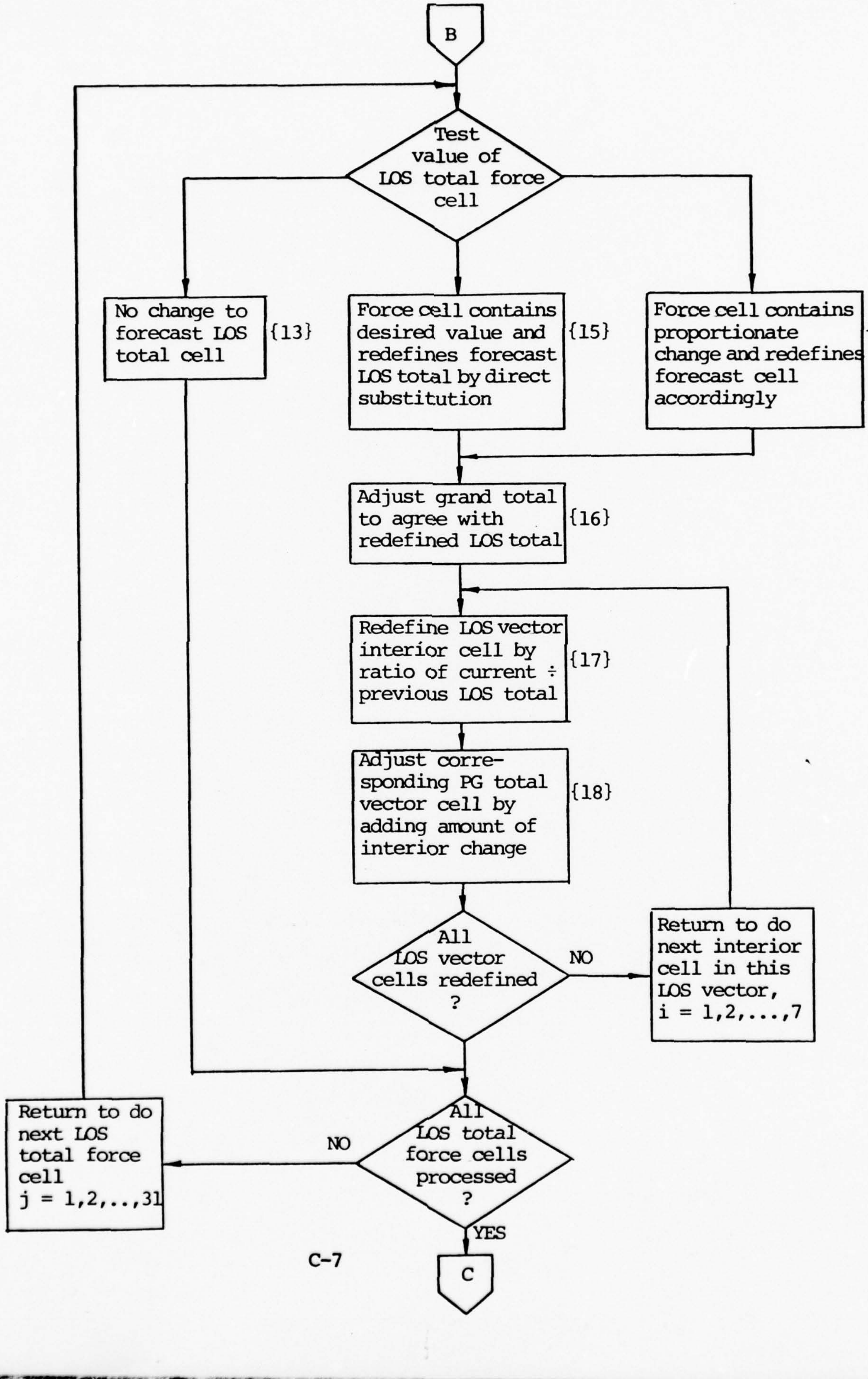
$$y_j^{(4)} = r \cdot y_j^{(3)} \tag{23}$$

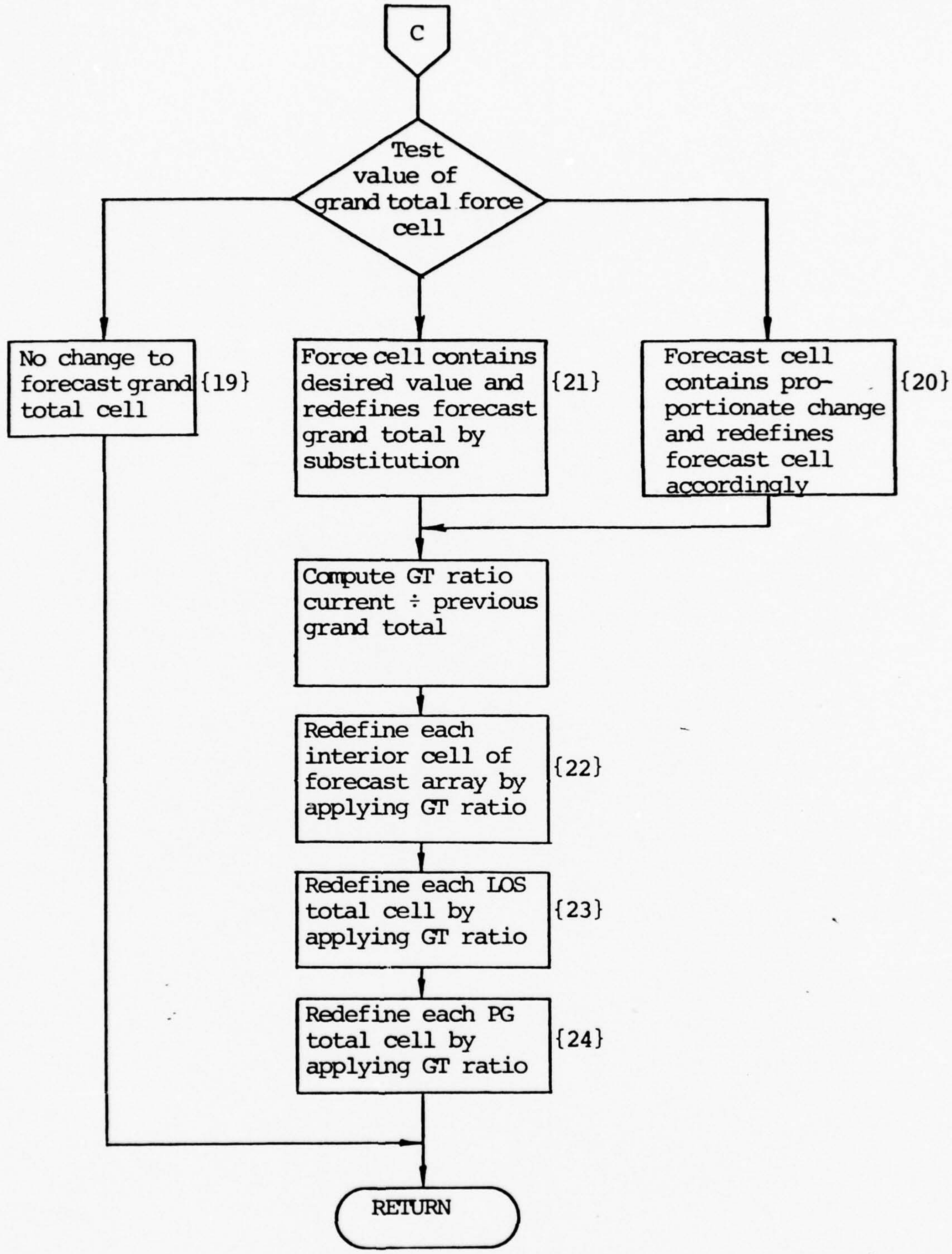
and $$p_i^{(4)} = r \cdot p_i^{(3)} \tag{24}$$

where $r = a^{(4)}/a^{(3)}$ $\quad i = 1,2,\ldots,7 \quad j = 1,2,\ldots,31$


APPFOR
INPUT: Force array, forecast array
Test value of interior force cell
No change to forecast interior cell
{1}
Force cell contains desired value and redefines forecast interior by direct substitution
{3}
Force cell contains proportionate change and redefines forecast cell accordingly
{2}
Adjust LOS total cell to agree with redefined interior
{4}
Adjust PG total cell to agree with redefined interior
{5}
Adjust grand total cell to agree with redefined interior
{6}
All interior cells processed ?
NO
YES
Return to do next interior force cell, i = 1,2,...,7 j = 1,2,..,31
A


A
Test value of PG total force cell
No change to forecast PG total cell
{7}
Force cell contains desired value and redefines forecast PG total by direct substitution
{9}
Force cell contains proportionate change and redefines forecast cell accordingly
{8}
Adjust grand total to agree with redefined PG total
{10}
Redefine PG vector interior cell by ratio of current ÷ previous PG total
{11}
Adjust corre- sponding LOS total vector cell by adding amount of interior change
{12}
All PG vector cells redefined ?
NO
Return to do next interior cell in this PG vector, j = 1,2,...,31
YES
All PG total force cells processed ?
Return to do next PG total force cell, i = 1,2,..,7
B


B
Test value of LOS total force cell
No change to forecast LOS total cell
{13}
Force cell contains desired value and redefines forecast LOS total by direct substitution
{15}
Force cell contains proportionate change and redefines forecast cell accordingly
{14}
Adjust grand total to agree with redefined LOS total
{16}
Redefine LOS vector interior cell by ratio of current ÷ previous LOS total
{17}
Adjust corresponding PG total vector cell by adding amount of interior change
{18}
All LOS vector cells redefined ?
NO
Return to do next interior cell in this LOS vector, i = 1,2,...,7
All LOS total force cells processed ?
NO
Return to do next LOS total force cell j = 1,2,..,31
YES
C


C
Test value of grand total force cell
No change to forecast grand total cell
{19}
Force cell contains desired value and redefines forecast grand total by substitution
{21}
Forecast cell contains proportionate change and redefines forecast cell accordingly
{20}
Compute GT ratio current ÷ previous grand total
Redefine each interior cell of forecast array by applying GT ratio
{22}
Redefine each LOS total cell by applying GT ratio
{23}
Redefine each PG total cell by applying GT ratio
{24}
RETURN

APPENDIX D

ROUND

B&K
DYNAMICS, INC.
15825 SHADY GROVE ROAD
ROCKVILLE, MARYLAND 20850

ROUND

The ROUND subroutine of FAST integerizes the cells of the forecast array and reestablishes the accord between vector total cells and the sum of the vector's component cells. The accord existing among the unrounded cells of the array is often disrupted by simple rounding.

The subroutine uses a common technique for computerized rounding to the nearest integer. The technique involves adding 0.5 to the value of the cell and truncating the sum to its greatest integer value.

ROUND first applies the rounding technique to each cell of the PG total vector.

$$p_i^{(1)} = \lfloor p_i + 0.5 \rfloor, \qquad \{1\}$$

representing the greatest integer value within the expression. These rounded PG total vector cells are now held constant, while the other components of the array are adjusted as necessary to bring about accord.

The interior cells are then rounded.

$$w_{ij}^{(1)} = \lfloor w_{ij} + 0.5 \rfloor \qquad \{2\}$$

The rounded interior cells are then summed along each PG vector to derive $p^{(tw)}$, temporary computational values of the PG total cells which do not redefine $p_i^{(1)}$.

$$p_i^{(tw)} = \sum_{j=1}^{31} w_{ij}^{(1)} \qquad \text{for } i = 1,2,\ldots,7 \qquad \{3\}$$

The required adjustment, n, along one PG vector is computed

$$n = p_i^{(1)} - p_i^{(tw)} \qquad \{4\}$$

If n = 0, no adjustment is required and the routine proceeds to compute the adjustment along the next PG vector.

If $n \neq 0$, an adjustment is required. The routine randomly selects an LOS cell within this PG vector using the unrounded PG vector as a weighted probability function. If the required adjustment is positive, the routine adds 1.0 to the selected cell. If the required adjustment is negative and the current value of the selected cell is positive, the routine subtracts 1.0 from the selected cell. If the selected cell is zero, the routine makes another random selection. If the routine selects ten zero cells in succession, the PG total cell is modified by adding 1.0.

$$w_{ij}^{(2)} = \begin{cases} w_{ij}^{(1)} + 1.0, & n > 0 \qquad \{5\} \\ w_{ij}^{(1)} - 1.0, & n < 0 \text{ and } w_{ij}^{(1)} > 0 \qquad \{6\} \end{cases}$$

$$p_i^{(1)} = p_i + 1.0, \qquad n < 0 \text{ and } w_{ij}^{(1)} = 0 \qquad \{7\}$$

This process of making unit adjustments to randomly selected interior cells is repeated until n adjustments have been made.

Upon completion of the n adjustments for a given PG vector, the routine continues to compute and process adjustments to the next PG vector, until all seven PG vectors have been adjusted.

Upon completion of the adjustments to all PG vectors, the LOS total vector cells are redefined as the sum of the

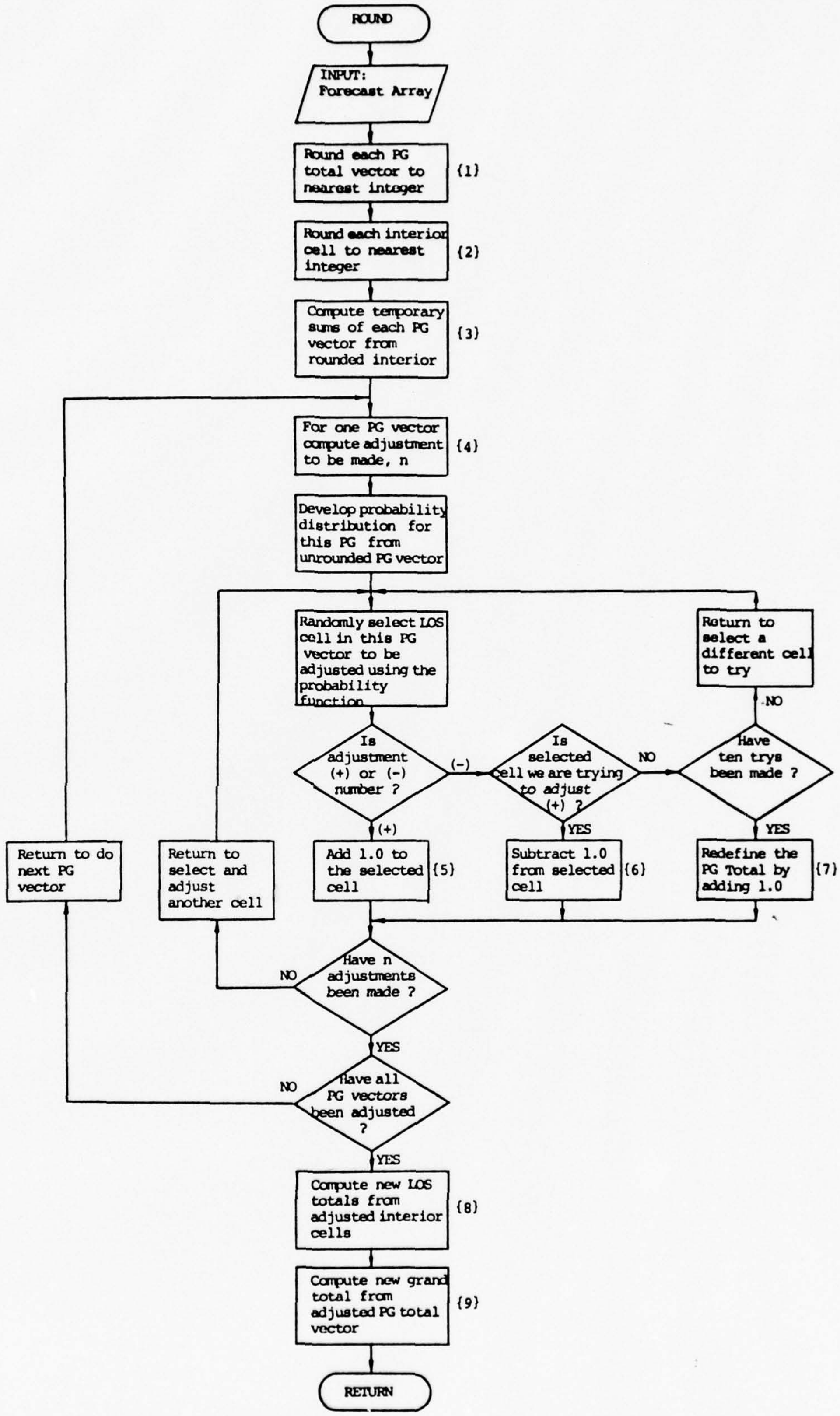
rounded and adjusted interior cells.

$$y_j^{(1)} = \sum_{i=1}^{7} w_{ij}^{(2)} \qquad \text{for } j = 1,2,\ldots,31 \qquad \{8\}$$

The grand total is then redefined as the sum of the (adjusted) PG total vector.

$$a^{(1)} = \sum_{i=1}^{7} p_i^{(1)} \qquad \{9\}$$

The array is now rounded and has total accord among its components.


ROUND
INPUT: Forecast Array
Round each PG total vector to nearest integer
{1}
Round each interior cell to nearest integer
{2}
Compute temporary sums of each PG vector from rounded interior
{3}
For one PG vector compute adjustment to be made, n
{4}
Develop probability distribution for this PG from unrounded PG vector
Randomly select LOS cell in this PG vector to be adjusted using the probability function
Return to select a different cell to try
NO
Is adjustment (+) or (-) number ?
(-)
Is selected cell we are trying to adjust (+) ?
NO
Have ten trys been made ?
(+)
YES
YES
Return to do next PG vector
Return to select and adjust another cell
Add 1.0 to the selected cell
{5}
Subtract 1.0 from selected cell
{6}
Redefine the PG Total by adding 1.0
{7}
Have n adjustments been made ?
NO
YES
Have all PG vectors been adjusted ?
NO
YES
Compute new LOS totals from adjusted interior cells
{8}
Compute new grand total from adjusted PG total vector
{9}
RETURN

APPENDIX E

ARAB

B&K DYNAMICS, INC.
15825 SHADY GROVE ROAD
ROCKVILLE, MARYLAND 20850

ARAB

The subroutine ARAB imposes the constraint that no cell in the forecast array may be greater than the corresponding cell in the base array, s_{ij}.

The routine first compares the PG total vector cells, i.e.: $p_i > s_{i,32}$ for $i = 1,2,\ldots,7$. If any forecast PG total cell is greater than the base cell, a diagnostic is printed and a return to the calling program is executed.

If all PG total cells pass the test, the routine continues by comparing the LOS total cells, i.e.: $y_j > s_{8,j}$ for $j = 1,2,\ldots,31$. If any forecast LOS total cell is greater than the corresponding base cell, a diagnostic is printed and a return to the calling program is executed.

If all the LOS total cells pass the test, the grand total cells are compared, i.e.: $a > s_{8,32}$.

If the grand total cell passes this test, the interior cells are compared cell-by-cell, i.e.: $w_{ij} > s_{ij}$ for $i = 1,2,\ldots,7$ and $j = 1,2,\ldots,31$. If a forecast interior cell is found to be greater than the corresponding base cell, the required adjustment, n, is computed for that cell.

$$n = w_{ij} - s_{ij} \qquad \{1\}$$

The interior cell and all corresponding vectors are adjusted accordingly.

$$w_{ij}^{(1)} = w_{ij} - n \tag{2}$$

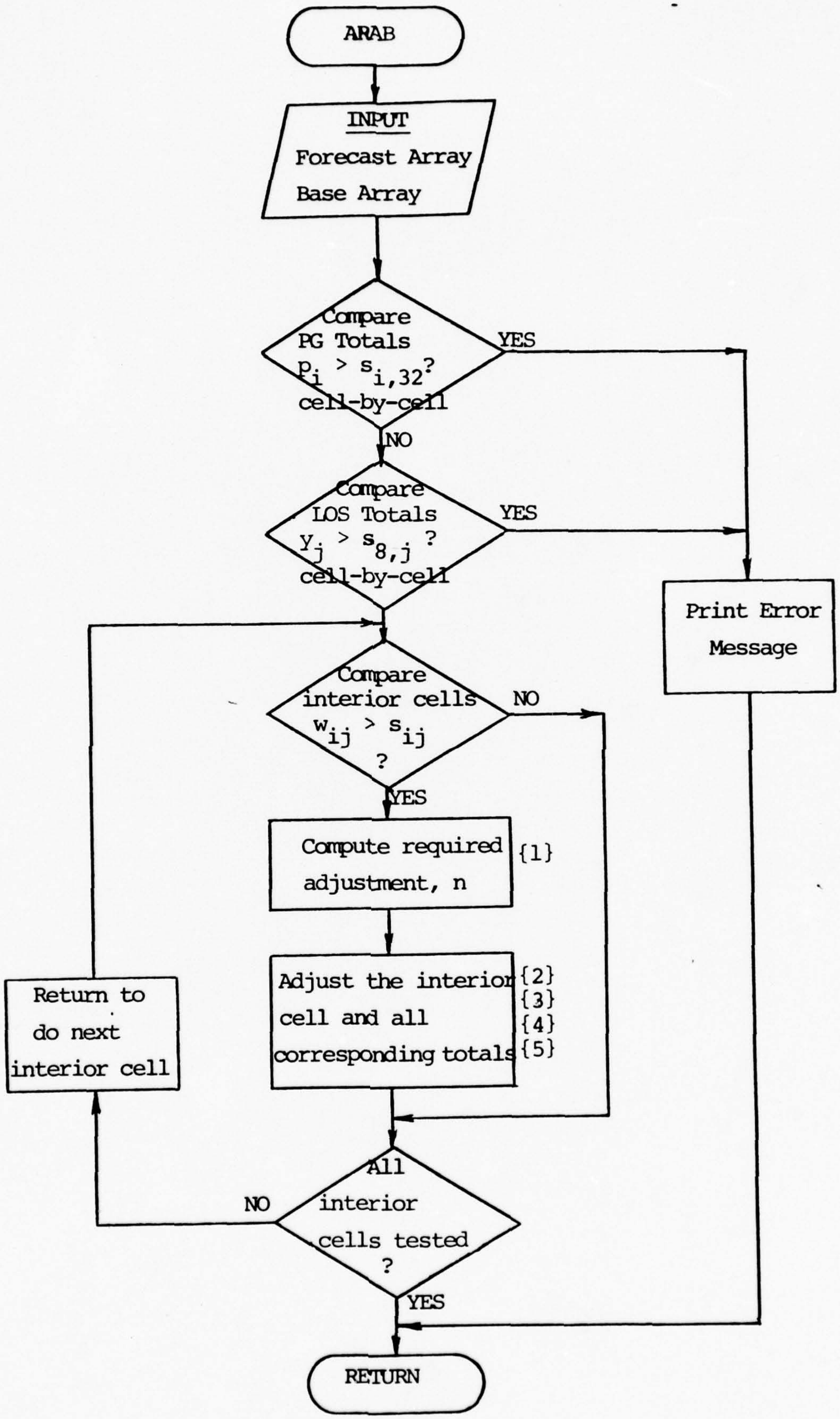
$$y_j^{(1)} = y_j - n \tag{3}$$

$$p_i^{(1)} = p_i - n \tag{4}$$

$$a^{(1)} = a - n \tag{5}$$

The process of testing, and possibly adjusting, is carried out for each interior cell.

Upon completion of the interior cell adjustments, ARAB executes a return to the calling program.


ARAB
INPUT
Forecast Array
Base Array
Compare
PG Totals
$p_i > s_{i,32}$?
cell-by-cell
YES
NO
Compare
LOS Totals
$y_j > s_{8,j}$?
cell-by-cell
YES
Print Error
Message
Compare
interior cells
$w_{ij} > s_{ij}$
?
NO
YES
Compute required
adjustment, n
{1}
Adjust the interior
cell and all
corresponding totals
{2}
{3}
{4}
{5}
Return to
do next
interior cell
All
interior
cells tested
?
NO
YES
RETURN

APPENDIX F

ARNG

B&K DYNAMICS, INC.
15825 SHADY GROVE ROAD
ROCKVILLE, MARYLAND 20850

ARNG

The ARNG subroutine of FAST is a monitor routine intended to control reconciliation of differences between cells of the ALNAV array and the summation of corresponding cells across all ratings for a given type of forecast array. The routine reconciles these differences while attempting to preserve the form of the predicted distributions across ratings for each cell of a forecast array.

The routine first locates the forecast arrays for all ratings having the forecast type being processed. The resulting data may be thought of as a three-dimensional array having coordinates for paygrade i, LOS j, and rating k. If, as a programmed function, the base constraint (see ARAB) is to be maintained, the routine also locates the corresponding base arrays. The routine then locates the ALNAV forecast array for this forecast type.

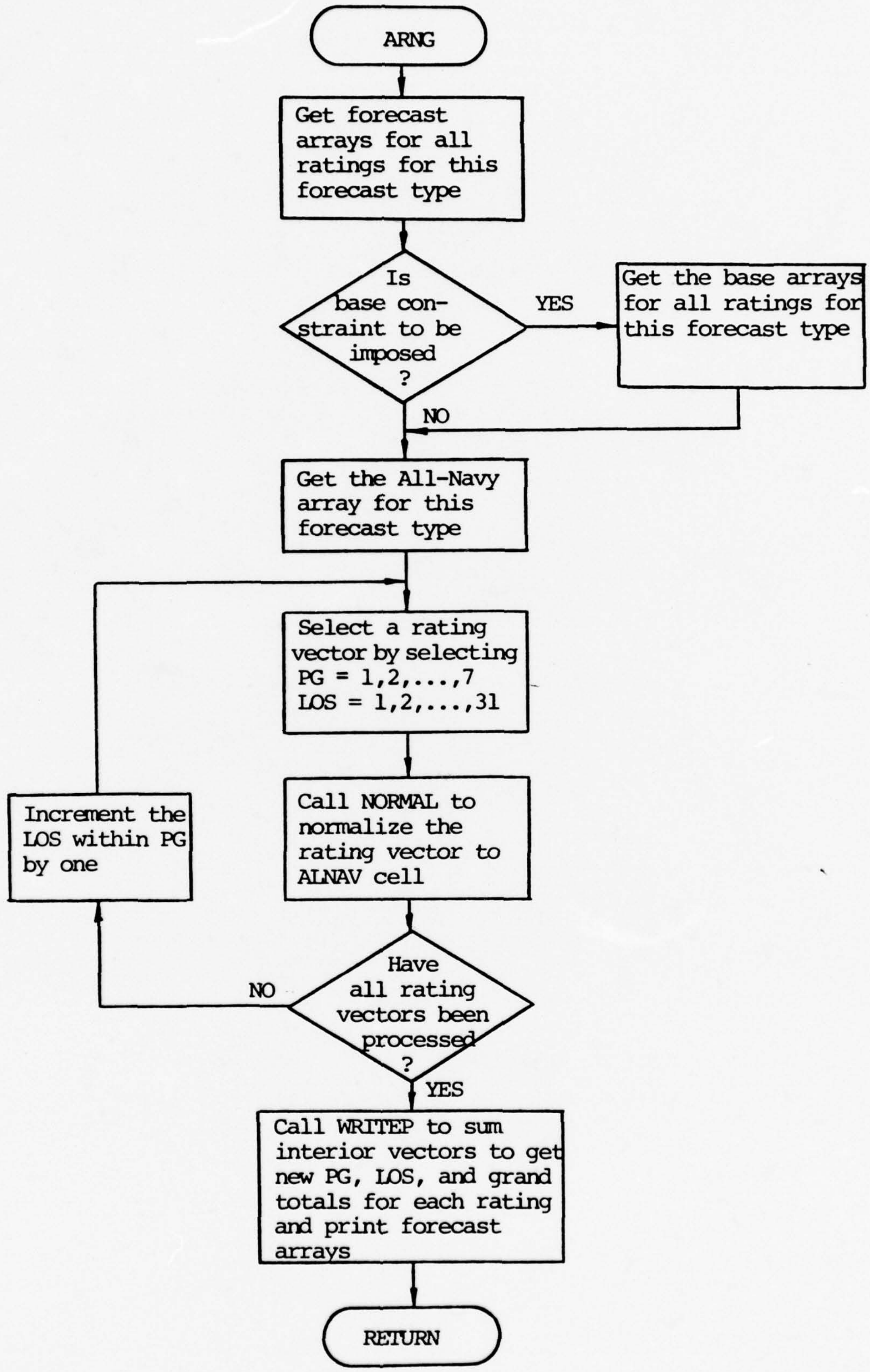
ARNG next steps cell-by-cell through the interior of the array. With each step, it selects a combination of paygrade (i = 1,2,...,7) and LOS (j = 1,2,...,31) which defines a rating vector (i,j) to be processed. Each step calls subroutine NORMAL to operate on the rating vector (i,j). NORMAL is the heart of ARNG and, essentially, performs a reconciliation of the differences between the predicted ALNAV value and the sum of the predicted rating vector. NORMAL involves a complex process for the selection of a method of reconciliation. NORMAL, due to its complexity, is described as a separate item.

NORMAL, when successfully executed, provides a redefined rating vector having a sum consistant with the corresponding cell in the ALNAV array, retains the contour of the original vector to the extent possible, and attempts to comply with the base constraint, if imposed.

Successive executions of NORMAL for each rating vector (i,j) for i = 1,2,...,7 and j = 1,2,...,31 results in possible redefinition of any or all interior cells, w_{ij}, of the forecast array for any rating, k.

Upon completion of the 217th execution of NORMAL, ARNG calls subroutine WRITEP. For each rating, k, subroutine WRITEP sums the existing interior cells of the rating's forecast array to establish new paygrade vector totals, new LOS vector totals, and a new grand total. These combine to form the final, corrected, forecast array for each rating within the given forecast (loss or gain) type. WRITEP optionally prints the final array. Upon completion of all ratings within the forecast type, WRITEP returns to ARNG.

ARNG, then, returns to FAST where the entire process re-initializes to operate upon the data for another forecast type.


ARNG
Get forecast arrays for all ratings for this forecast type
Is base constraint to be imposed ?
YES
Get the base arrays for all ratings for this forecast type
NO
Get the All-Navy array for this forecast type
Select a rating vector by selecting PG = 1,2,...,7 LOS = 1,2,...,31
Call NORMAL to normalize the rating vector to ALNAV cell
Increment the LOS within PG by one
Have all rating vectors been processed ?
NO
YES
Call WRITEP to sum interior vectors to get new PG, LOS, and grand totals for each rating and print forecast arrays
RETURN

APPENDIX G

NORMAL

B&K
DYNAMICS, INC.
15825 SHADY GROVE ROAD
ROCKVILLE, MARYLAND 20850

NORMAL

Subroutine NORMAL is called by ARNG to resolve inconsistencies between the depthwise summation of current values of forecast cells in a rating vector and the value in the corresponding cell of the ALNAV forecast array.

ARNG has provided, as input to NORMAL, a rating vector, w_k, containing forecast cells for paygrade i and LOS j for each rating k within a given forecast (loss or gain) type. ARNG has also provided a comparable rating vector for the prediction base, s_k, if the base constraint is to be imposed. The appropriate ALNAV forecast array cell, u, has also been provided to NORMAL and is considered by NORMAL to be the desired sum of the given rating vector.

NORMAL begins by computing u^w, the current value sum of the interior cells of the rating vector.

$$u^w = \sum_{k=1}^{n} w_k \qquad \text{, where n represents the number of ratings} \qquad \{1\}$$

Before continuing computations, NORMAL tests to determine if u, the desired rating vector total, is zero. If it is, the interior cells of the rating vector should also be zero. All cells of the rating vector are, therefore, set to equal zero and NORMAL returns to ARNG.

If the ALNAV cell, u, is not zero, NORMAL computes the required-adjustment, d, for this rating vector.

$$d = u - u^w \qquad \{2\}$$

If no adjustment is required, i.e.: d = 0, NORMAL returns to ARNG. If an adjustment is required, NORMAL continues.

NORMAL now enters an involved process of determining the method of distribution to be used in adjusting the interior cells of the rating vector to be consistant with the ALNAV cell. NORMAL attempts to make the adjustment required by increasing or decreasing the counts in a rating vector in integer units by means of a random distribution algorithm employing the original rating vector as a probability distribution function. Parameters for the distribution algorithm are selected according to combinations of the following criteria:

(1) Direction of the required adjustment (increase or decrease),

(2) Reliability of distributions,

(3) Availability of base data, and

(4) Opportunity to optimize computer costs

The selection of the dataset from which to develop the probability distribution follows a hierarchy of general desirability, as follows:

(1) Forecast rating vector with base constraint

(2) Forecast rating vector without base constraint

(3) Base data rating vector with base constraint

(4) Forecast paygrade total rating vector with no constraint

(5) Base data paygrade total rating vector with no constraint

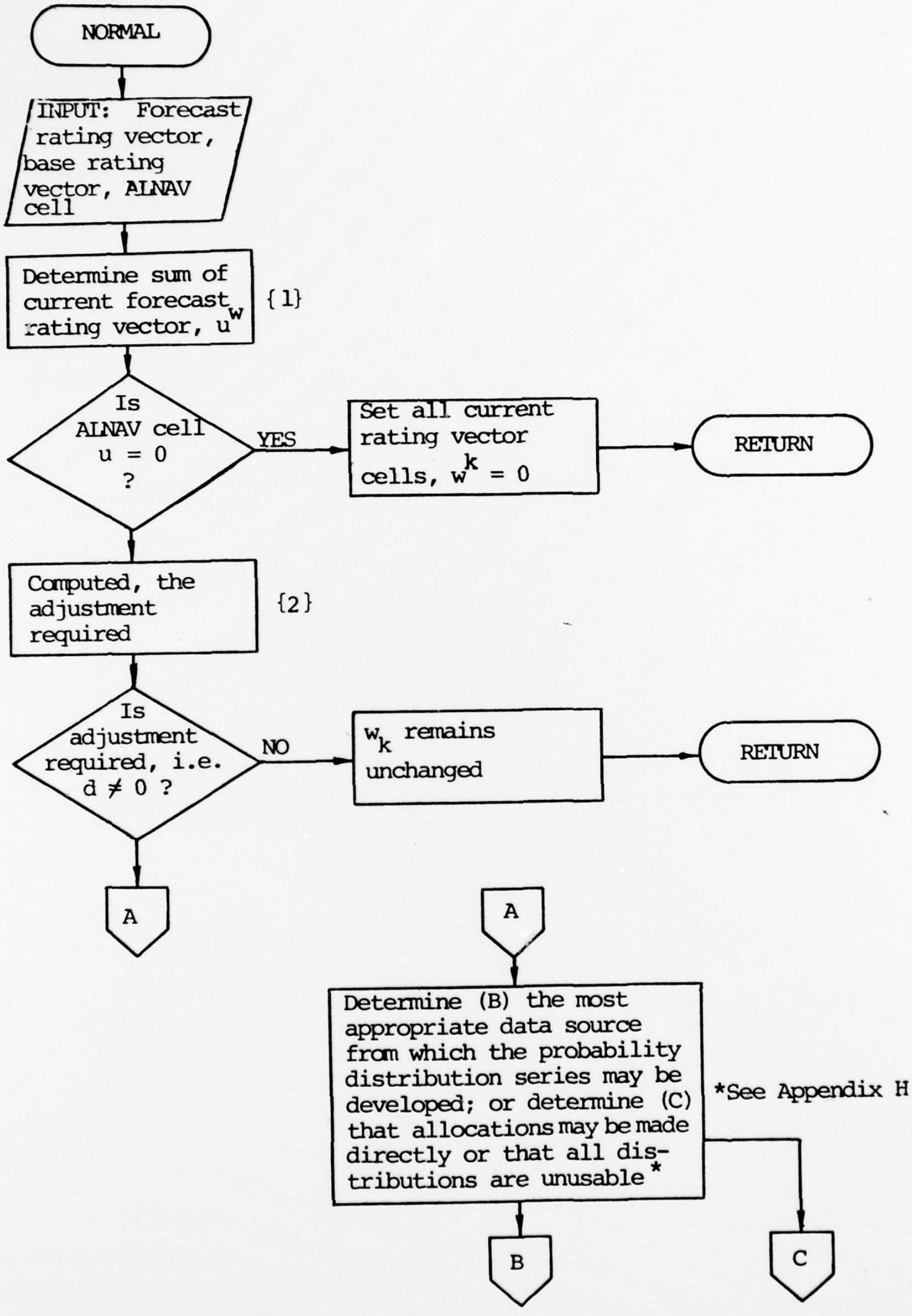
(6) Equal probability distribution, if permitted.

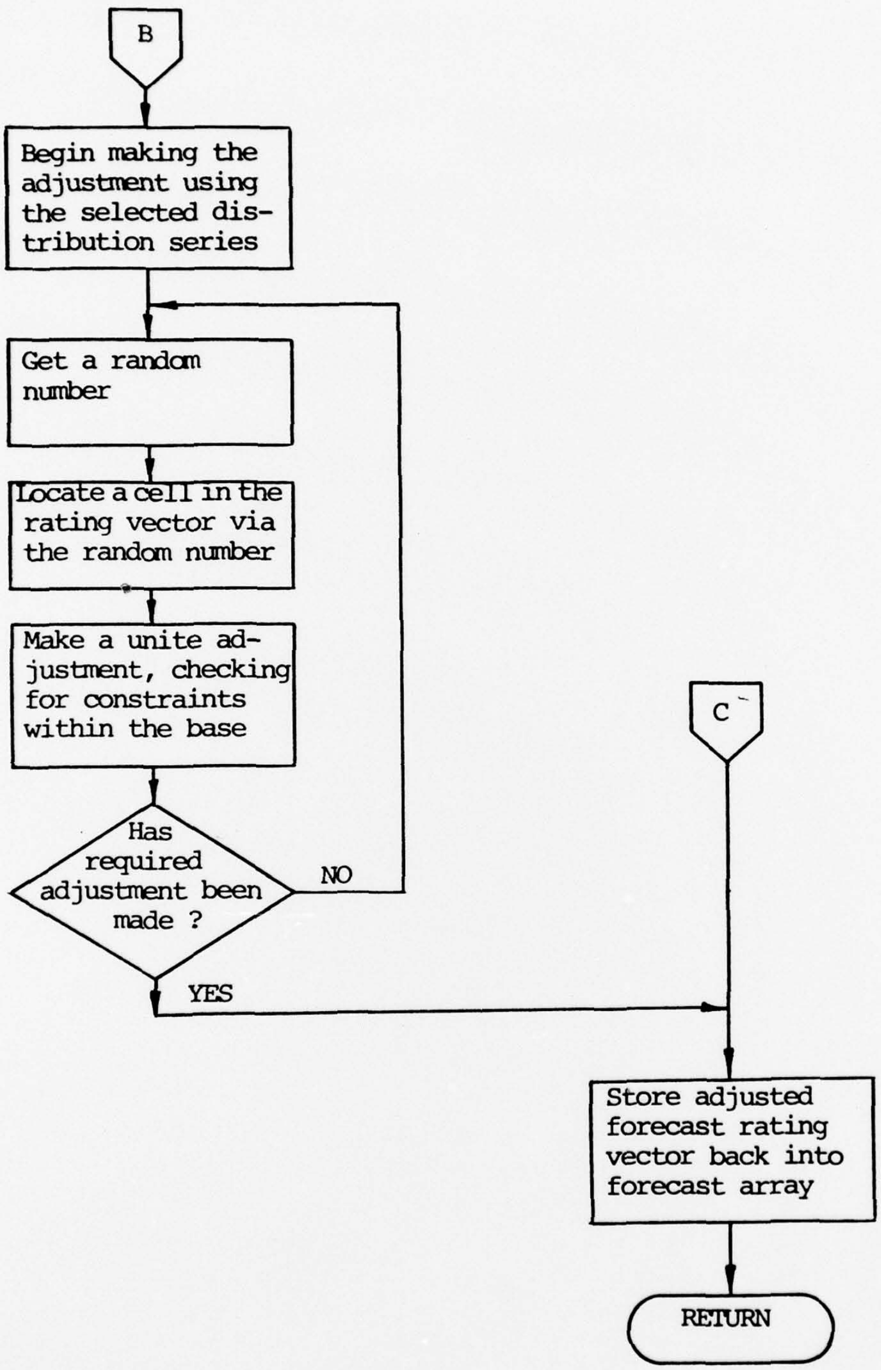
The process of selecting the data set from which to develop the distribution series is described in detail in Appendix H.

Once Normal has chosen the basis for deriving the distribution series and, possibly, redefined the required adjustment as the ALNAV total to facilitate negative adjustment for computer optimization, the program distributes the required adjustment

by making unit increases to randomly selected cells of the forecast rating vector, until d adjustments have been made. The process of making each unit adjustment involves: obtaining a random number; using it and the distribution series to select k, the rating of the forecast rating vector w_k; making a unit adjustment to w_k; and recycling to make further unit adjustments until the required adjustment has been completed. The sum of the populations of the cells in the adjusted forecast rating vector should now equal the corresponding ALNAV total cell.

The adjusted forecast rating vector is used to redefine the forecast array and NORMAL returns to ARNG.


NORMAL
INPUT: Forecast rating vector, base rating vector, ALNAV cell
Determine sum of current forecast rating vector, u^w
{1}
Is ALNAV cell u = 0 ?
YES
Set all current rating vector cells, w^k = 0
RETURN
Computed, the adjustment required
{2}
Is adjustment required, i.e. d ≠ 0 ?
NO
w_k remains unchanged
RETURN
A
A
Determine (B) the most appropriate data source from which the probability distribution series may be developed; or determine (C) that allocations may be made directly or that all distributions are unusable*
*See Appendix H
B
C


B
Begin making the adjustment using the selected distribution series
Get a random number
Locate a cell in the rating vector via the random number
Make a unite adjustment, checking for constraints within the base
Has required adjustment been made ?
NO
YES
C
Store adjusted forecast rating vector back into forecast array
RETURN

APPENDIX H

Probability Distribution Function
Selection Within NORMAL

B&K
DYNAMICS, INC.
15825 SHADY GROVE ROAD
ROCKVILLE, MARYLAND 20850

APPENDIX A

Probability Distribution Function Selection Within NORMAL

The process of selecting the most appropriate data series to serve as the basis for developing a probability distribution function within the NORMAL subroutine is described in the following. The process is primarily a series of decision logic and, from each decision point, a given alternative is followed to an end conclusion prior to pursuit of the other alternative(s). Persistent referral to the accompanying flow diagram will aid in comprehension of this text.

(1) The process first tests the previously derived value of d, the required adjustment, to determine if it is negative. A negative value indicates that the population in the forecast rating vector must decrease to reconcile it with the ALNAV cell.

(2) If it is determined that d is negative, the process attempts to optimize computer efficiency by recognizing that the relative adjustment is very small and avoiding excessive iteration through the random number generation. The relative adjustment, z, is computed as:

$$z = u/u^{w} \tag{3}$$

Since it has been determined that $u < u^{w}$, it follows that $z < 1.0$. It is a programmed function that if the ALNAV total is less than 5% of the sum of the forecast rating vector population, or if the ALNAV total is very small ($u \leq 2$), it will be more efficient to use a reverse random distribution technique.

(3) The reverse random distribution technique follows the theory that for very small relative adjustments it will be more efficient to simply distribute the ALNAV total randomly across the rating vector, rather than distribute the required-adjustment among the existing forecast rating vector cells. The execution of this reverse random distribution involves: (a) Developing a distribution series from the existing forecast rating vector, (b) Equating the contents of the forecast rating vector to zero, (c) Equating the required-adjustment to the population of the ALNAV total, and (d) Distributing the redefined required-adjustment to randomly selected cells of the cleared forecast rating vector. The reverse random distribution may be expressed as:

If $z < .05$ or if $u \leq 2$, then: {4}

(a) $(\text{distribution series})_m = \sum_{k=1}^{m} w_k$

where $m = 1,2,\ldots,$number of ratings

(b) $w_k = 0$, where $k = 1,2,\ldots,$number of ratings

(c) $d^{(1)} = u$

(4) If it was determined in paragraph (2) that the reverse distribution is not a worthwhile optimization, a normal random distribution of the required-adjustment to cells of the forecast rating vector is executed. The forecast rating vector is used as the basis for developing the distribution function.

(5) If, in paragraph (1), it is determined that the required-adjustment is positive, the process then tests to determine whether or not the base constraint is to be maintained. The base constraint, as imposed in subroutine ARAB, provides that no cell population within the rating vector may exceed the ALNAV total, u. If no attempt to maintain the base

constraint is to be made, the program continues as described in paragraph (6). If the base constraint is to be maintained, the program continues as described in paragraph (9).

(6) If no attempt to maintain the base constraint is required, NORMAL tries to use the forecast rating vector to develop the distribution function. A valid distribution function may be developed if an adequate number of cells (currently five or more) in the rating vector are populated ($\neq 0$). NORMAL, therefore, counts the populated cells in the forecast rating vector to determine adequacy. If sufficient cells of the forecast rating vector are populated, it is used to develop the distribution function for the adjustment process.

(7) If, in paragrpah (6), the forecast rating vector was found to have insufficient populated cells to provide a valid distribution function, NORMAL continues to search for a valid basis for the distribution function by trying the forecast PG total rating vector. As with the forecast rating vector, the number of populated cells in the forecast PG total rating vector are counted. If there are sufficient populated cells (five or more), the program uses the forecast PG total rating vector to develop the distribution function for the adjustment process.

(8) If as in paragraph (7), the forecast PG total rating vector is determined to be inadequate for the purpose, the program tests a user defined abort switch parameter to choose between the use of an equal probability distribution or a direct abort of the attempt to reconcile the vector to the ALNAV cell.

(9) If, as in paragraph (5), it is determined that the base constraint is to be maintained, the program first tries to use the forecast rating vector to develop the distribution function. As described in paragraph (6), the vector must have a sufficient number of populated cells to be adequate as the basis for the distribution function. If it is determined to be adequate, the program computes, e^w, the adjustment-potential of the forecast rating vector.

$$e^w = \sum_{k=1}^{n} (s_k - w_k) \quad \text{except where } w_k = 0 \qquad \{5\}$$

where: n = number of ratings, and

w_k = cells in the forecast rating vector

The adjustment-potential is then compared to the required-adjustment, d.

(10) If the adjustment-potential of the forecast rating vector equals the required-adjustment, the adjustments are allocated directly.

If $e^w = d$, then:

$$w_k^{(1)} = s_k, \quad \text{where } k = 1,2,\ldots, \text{number of ratings, except where } w_k = 0 \qquad \{6\}$$

(11) If, as in paragraph (9), the adjustment-potential is found to be greater than the required-adjustment, the program uses the forecast rating vector to develop the distribution function. Prior to developing the function, the program attempts to optimize computer efficiency by trying to avoid excessive iteration through the random number generation. The relative adjustment, z, is computed as the ratio of the required-adjustment to the adjustment-potential, i.e.:

$$z = d/e^w \qquad \{7\}$$

If the required-adjustment is 95% of the adjustment-potential, $z > .95$, or if the difference between the potential and the required adjustment is small, $(e^w - d) < 2$, the program optimizes computer cost by applying the reverse random distribution technique described in paragraph (3).

(12) If, as in paragraph (9), the adjustment-potential of the forecast rating vector is found to be less than the required-adjustment, the program tries to use the base rating vector. The adjustment potential, e^s, of the base rating vector is determined.

$$e^s = \sum_{k-1}^{n} (s_k - w_k) \quad \text{except where } w_k = 0 \qquad \{8\}$$

where: n = number of ratings

w_k = cells in base rating vector

The adjustment-potential of the base rating vector is then compared to the required-adjustment.

(13) If the adjustment potential of the base rating vector equals the required-adjustment, the adjustments are allocated directly.

If $e^w = d$, then:

$$w_k^{(1)} = s_k, \text{ where } k = 1,2,\ldots, \text{ number of ratings, except where } w_k = 0 \qquad \{9\}$$

(14) If, as in paragraph (12), the adjustment-potential of the base rating vector is found to be greater than the required-adjustment, the program uses the vector to develop the distribution series. Prior to developing the function, the program attempts to optimize computer efficiency by the means described in paragraph (9), using the formula:

$$z = d/e^s \qquad \{10\}$$

If $z > .95$ or if $(e^s - d) < 2$, the program optimizes computer efficiency by applying the reverse random distribution technique described in paragraph (3).

(15) If, as in paragraph (12), it is determined that the adjustment-potential of the base rating vector is less than the required-adjustment, the program uses the forecast PG total rating vector to develop the distribution function.

(16) If, as in paragraph (9), it is determined that the forecast rating vector is inadequate, the program uses the rating vector to develop the distribution function. The number of populated cells in the base rating vector are determined.

(17) If, the number of populated cells in the base rating vector prove to be sufficient, the adjustment potential of the vector is computed.

$$e^s = \sum_{k=1}^{n} (s_k - w_k), \quad \text{except where } w_k = 0 \qquad \{11\}$$

where: n = number of ratings

w_k = cells in base rating vector.

(18) The adjustment-potential of the base rating vector is compared to the required adjustment. If the adjustment-potential of the base rating vector equals the required-adjustment, adjustments are allocated directly.

If $e^s = d$, then

$$w_k^{(1)} = s_k, \quad \text{where } k = 1,2,\ldots, \text{ number of ratings,} \qquad \{12\}$$
$$\text{except where } w_k = 0$$

(19) If, as in paragraph (17), the adjustment-potential is determined to be greater than the required-adjustment, the base rating vector is used to develop the distribution function. Prior to developing the function, the program attempts to optimize computer efficiency by the means described in paragraph (9), using the formula:

$$z = d/e^S \qquad \{13\}$$

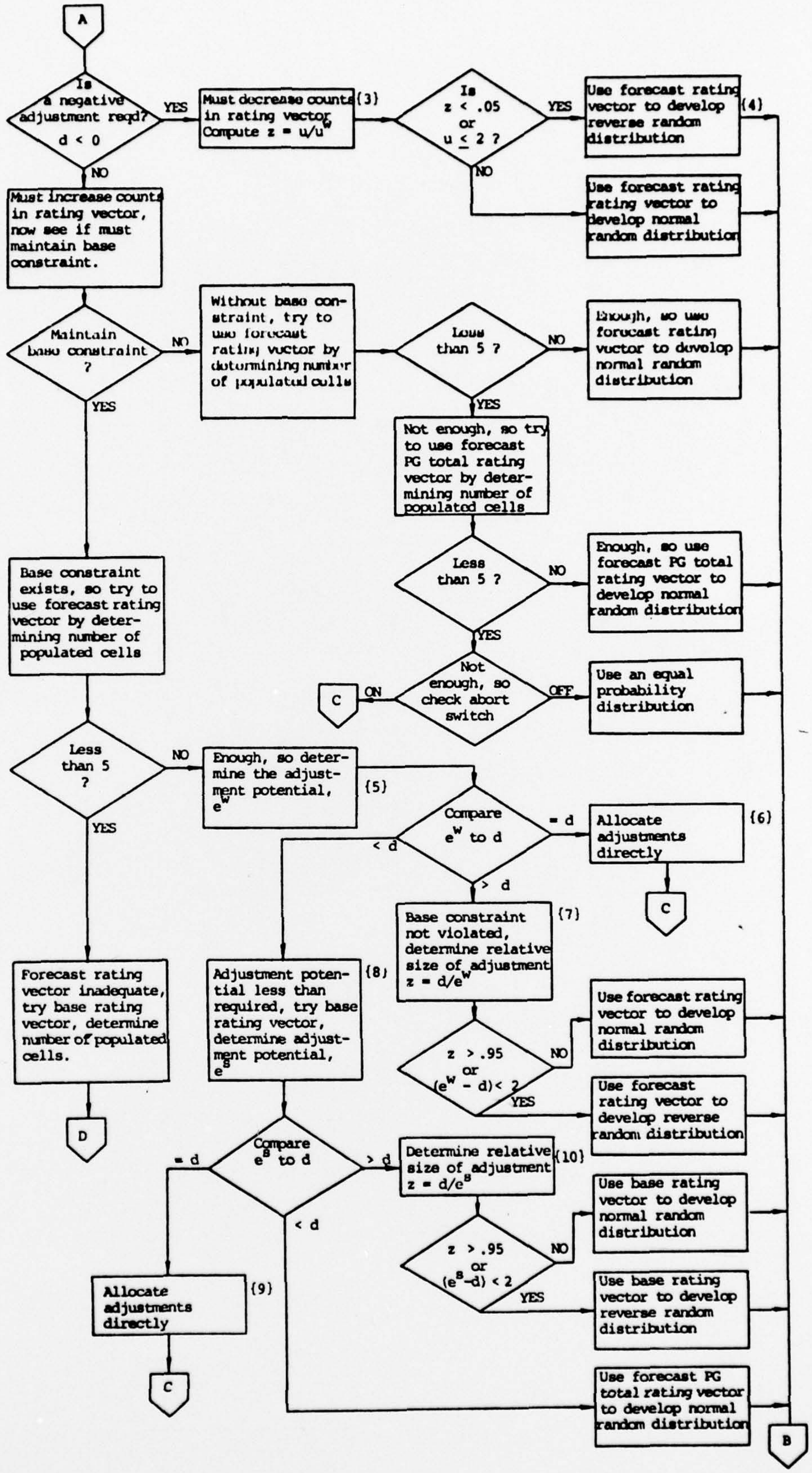
If $z > .95$ or if $(e^S - d) < 2$, the program optimizes computer efficiency by applying the reverse random distribution technique described in paragraph (3).

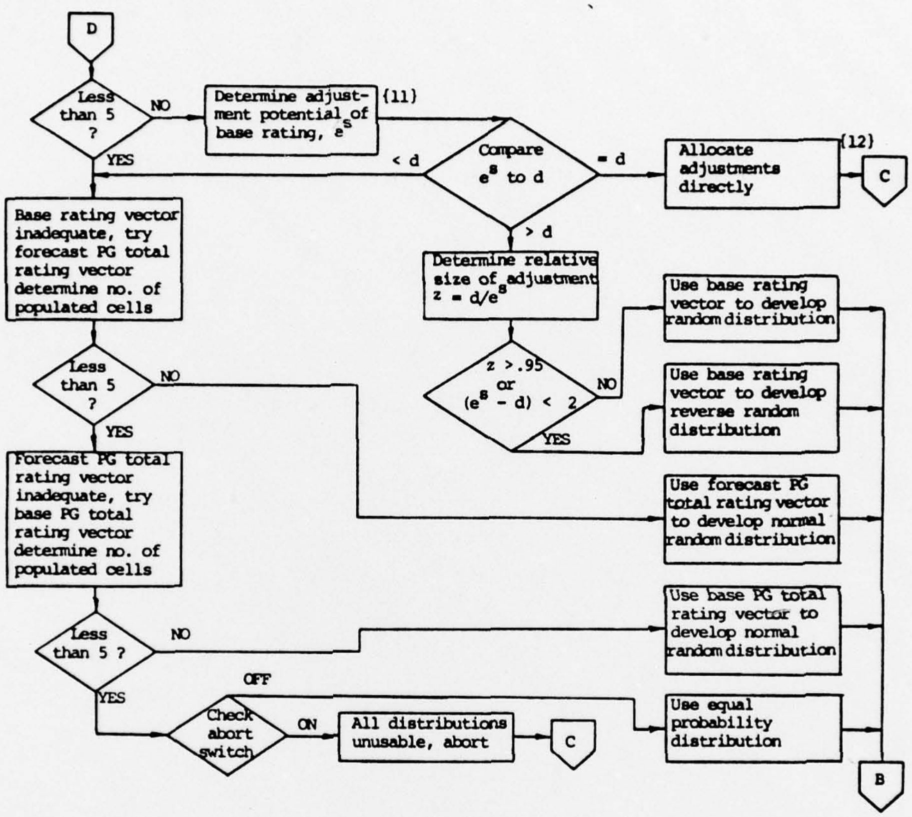
(20) If, as in paragraph (17), the adjustment-potential is determined to be less than the required-adjustment; or if, as in paragraph (16), the number of populated cells in the base rating vector are determined to be insufficient to provide an adequate basis for developing the distribution funciton; then, the program attempts to use the forecast paygrade total rating vector.

(21) If the number of populated cells in the forecast PG total rating vector are adequate, it is used to develop the distribution function.

(22) If the number of populated cells in the forecast PG total rating vector are inadequate, the program attempts to use the base PG total rating vector to develop the distribution function.

(23) If the number of populated cells in the base PG total rating vector are insufficient to provide an adequate basis from which to develop the distribution funciton, the program tests the suer defined abort switch. This switch determines whether the program resprts to use of equal probability distribution, or aborts the attempt to reconcile the forecast rating vector to the ALNAV cell.


A
Is a negative adjustment reqd? d < 0
YES
NO
Must decrease counts in rating vector Compute z = u/u^w
{3}
Is z < .05 or u ≤ 2 ?
YES
NO
Use forecast rating vector to develop reverse random distribution
{4}
Use forecast rating rating vector to develop normal random distribution
Must increase counts in rating vector, now see if must maintain base constraint.
Maintain base constraint ?
NO
YES
Without base constraint, try to use forecast rating vector by determining number of populated cells
Less than 5 ?
NO
YES
Enough, so use forecast rating vector to develop normal random distribution
Not enough, so try to use forecast PG total rating vector by determining number of populated cells
Less than 5 ?
NO
YES
Enough, so use forecast PG total rating vector to develop normal random distribution
Not enough, so check abort switch
ON
C
OFF
Use an equal probability distribution
Base constraint exists, so try to use forecast rating vector by determining number of populated cells
Less than 5 ?
NO
YES
Enough, so determine the adjustment potential, e^w
{5}
Compare e^w to d
= d
< d
> d
Allocate adjustments directly
{6}
C
Base constraint not violated, determine relative size of adjustment z = d/e^w
{7}
z > .95 or (e^w − d) < 2
NO
YES
Use forecast rating vector to develop normal random distribution
Use forecast rating vector to develop reverse random distribution
Forecast rating vector inadequate, try base rating vector, determine number of populated cells.
D
Adjustment potential less than required, try base rating vector, determine adjustment potential, e^B
{8}
Compare e^B to d
= d
> d
< d
Determine relative size of adjustment z = d/e^B
{10}
z > .95 or (e^B−d) < 2
NO
YES
Use base rating vector to develop normal random distribution
Use base rating vector to develop reverse random distribution
Allocate adjustments directly
{9}
C
Use forecast PG total rating vector to develop normal random distribution
B


D
Less than 5 ?
NO
YES
Determine adjustment potential of base rating, e^s
{11}
Compare e^s to d
< d
= d
> d
Allocate adjustments directly
{12}
C
Base rating vector inadequate, try forecast PG total rating vector determine no. of populated cells
Determine relative size of adjustment $z = d/e^s$
Use base rating vector to develop random distribution
Less than 5 ?
NO
YES
z > .95 or $(e^s - d) < 2$
NO
YES
Use base rating vector to develop reverse random distribution
Forecast PG total rating vector inadequate, try base PG total rating vector determine no. of populated cells
Use forecast PG total rating vector to develop normal random distribution
Less than 5 ?
NO
YES
Use base PG total rating vector to develop normal random distribution
OFF
Check abort switch
ON
All distributions unusable, abort
C
Use equal probability distribution
B